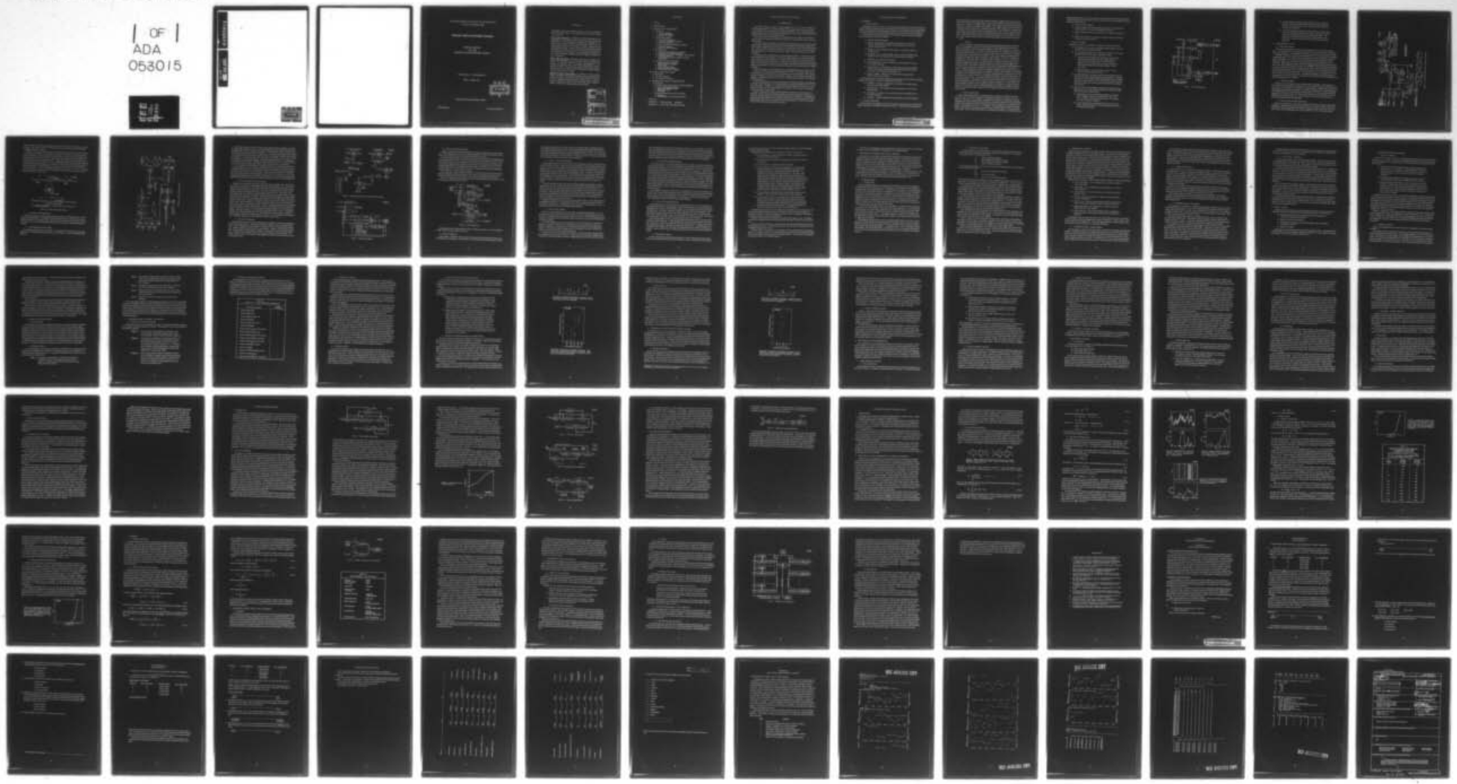
AD-A053 015 MASSACHUSETTS INST OF TECH LEXINGTON LINCOLN LAB F/G 9/4
NETWORK SPEECH PROCESSING PROGRAM. (U)
SEP 77 M A HERLIN, T BIALLY F19628-76-C-0002
UNCLASSIFIED ESD-TR-77-337 NL

1 OF 1
ADA
053015



END
DATE
FILMED
6 -78
DDC

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

NETWORK SPEECH PROCESSING PROGRAM

ANNUAL REPORT
TO THE
DEFENSE COMMUNICATIONS AGENCY

1 OCTOBER 1976 – 30 SEPTEMBER 1977

ISSUED 2 MARCH 1978

Approved for public release; distribution unlimited.

LEXINGTON MASSACHUSETTS

ABSTRACT

This report covers work performed during FY 1977 on the DCA Network Speech Processing Contract. Three general areas of work are reported in this document.

Secure Voice Conferencing: This effort consisted of the design and implementation of a flexible test facility that can accommodate up to twenty participants in a wide variety of voice-conferencing configurations, the design of human factors methods for the quantitative evaluation of voice-conferencing techniques, and the conduct of actual conferencing tests using trained personnel. Included in this report are preliminary results on the effects of conference size and delay on problem-solving efficiency in a teleconferencing environment, and recommendations regarding the use of signal summation and signal selection techniques in military secure voice systems.

Speech Algorithm Studies: This report describes an automatic-gain-control (AGC) algorithm for use with the Linear Predictive Coding (LPC) system. The algorithm extends the dynamic range capability of LPC vocoders by a significant factor, and is easily implemented in a microprocessor terminal. Also described is an experiment aimed at improving the quality of LPC speech via the introduction of a channel-vocoded version of the residual error signal.

Bandwidth Efficient Communications: The behavior of voice and data traffic in the Slotted Envelope Network (SENET) has been determined via computer simulation, and results indicate a strong need for flow control in order to maintain reasonable data buffer sizes and delays. Requirements for a packetized virtual circuit (PVC) nodal switch are also included, in addition to a proposed hardware switch design.

CONTENTS

NETWORK SPEECH PROCESSING

I. INTRODUCTION

This report documents work at Lincoln Laboratory in FY 1977 on the DCA-sponsored Network Speech Program. The effort consisted of three major tasks focusing on (1) secure voice conferencing, (2) narrowband speech digitizing algorithms, and (3) bandwidth efficient communications. Each of these tasks was directed at particular problems associated with the design of future defense communications systems.

The secure-voice-conferencing effort was concerned with the analysis and simulation of various conference bridging and switching configurations, control protocols, delays, and conference sizes.

The effort in speech algorithms was primarily directed toward the severe problem of wide speaker dynamic range which causes distortion in narrowband speech digitizers (specifically LPC vocoders). Solutions to this problem will also enhance the quality of speech realized from LPC-CVSD and CVSD-LPC tandem connections. Some research on LPC error signal coding was also performed, as reported in the Network Speech Processing Semiannual Technical Summary.[1]

The ongoing study of bandwidth efficient communications systems, in particular the packetized virtual circuit (PVC) approach, has yielded valuable data on circuit and system utilization and efficiency, network delays, and various sources of distortion, as well as an initial look at PVC nodal switch requirements. To learn more about the effects of fluctuations in the number of voice users in combined voice data networks, the Slotted Envelope Network (SENET)[2] system was studied and some of its performance characteristics were determined via computer simulation.

Sections II through IV describe progress and conclusions about the three technical areas.

In Sec. II, the secure-voice-conferencing effort is reviewed. The approach taken by Lincoln Laboratory to quantify answers to a broad spectrum of critical questions in the conferencing area is discussed, and details of the hardware, software, and human factors approach are presented. Conclusions and answers based on FY 77 experiments, simulations, and analyses are also presented.

Section III presents the concise but important study of combating dynamic range distortion in the LPC vocoder algorithm, and the final algorithm modifications proposed for the LL-DCA LPCM experiment. In addition, a concept for improving the quality of LPC speech via the inclusion of a channel-vocoded residual signal is summarized, and the details of an attempted real-time simulation are described.

Section IV deals with the effort in bandwidth efficient combined voice-data networks. In particular, this section presents additional results of the Lincoln Laboratory-proposed PVC voice-data network analysis as well as a possible nodal switch architecture. A simulation and analysis of the SENET network are presented in this section to provide insights into the effect of fluctuations in the number of voice users on the network efficiency, since the PVC studies did not consider this aspect of parameter variations.

II. SECURE VOICE CONFERENCING

A. OVERVIEW

1. Problem Definition

Lincoln Laboratory's FY 77 effort in secure voice conferencing was directed at (a) analyzing various methods of achieving secure voice conferencing, (b) providing appropriate demonstrations of the most promising of these techniques, and, finally, (c) recommending the best method of secure voice conferencing for ultimate incorporation into worldwide secure voice architectures.

Currently known conferencing methods can be divided into signal-summation and signal-selection (broadcast) techniques as follows:

a. Signal-Summation Techniques

(1) Analog – analog signals are summed and transmitted to all receivers. This is considered to be a baseline technique against which other approaches can be compared.

(2) Digital – for frame-oriented vocoder signals, a combining of signals at the digital level.

(3) Voting logic – for bit-oriented signals such as CVSD, the method of mapping pairs of bits into slopes and then majority voting on the slopes.

b. Signal-Selection (Broadcast) Techniques

(1) Control algorithms – structured conferencing employing either a human or a computer chairman.

(2) Voice detection vs control signals – either automatic detection of speaker activity or touch-tone or push-to-talk inputs by conference participants can be used for determining who will speak next.

(3) Interrupt methods – means for informing the broadcast speaker that another conferee wants the floor.

Our efforts have been directed at studying and analyzing the above-described conferencing techniques with respect to a variety of critical issues, including:

(a) Flexibility – the degree of adaptability and extension into unplanned areas for each conferencing technique.

(b) Conference size – size limitations and effects inherent in various conferencing methods.

(c) Voice quality – the effects of voice-quality limitations resulting from use of specific techniques.

(d) Complexity – relative implementation complexities of the various methods.

2. Research Methodology

Given the extent and complexity of many of the technical requirements, it was clear that a unified, systematic, experimental approach was in order. To proceed with this approach, a facility was designed on which all of the pertinent conference configurations and control structures could be set up with flexible software, so that a set of conferees could dial in and use the facility for a conference. In addition to the facility, human factors experts from Bolt Beranek & Newman, Inc. (BBN) of Cambridge, Massachusetts were brought in under contract to provide experimental design and human factors evaluation expertise. This combination of a flexible hardware facility on which various conference techniques can be set up, and human factors expertise for the design and evaluation of conferencing experiments has been producing valuable data on specific conferencing configurations for the past several months of the FY 77 effort. The facilities design and human factors efforts are outlined briefly below and elaborated upon later in this report.

a. Facility

In order to implement the large set of disparate conferencing configurations required for the FY 77 investigations, a flexible conferencing facility was designed around a touch-tone dial-up capability and a signal processing switchboard. Rather than hardwire 15 to 20 phone lines into a switching array, we decided to provide a flexible dial-up capability so that any location within Lincoln Laboratory as well as several distant users (e.g., DCEC-Reston, Virginia) could participate in these conference experiments. The receiver modems for answering and decoding of touch-tone information can inform a central computer (PDP 11/45) of requests to participate in a conference leave a conference, speak, interrupt, and so on. In this way, the facility keeps track of conferees. The actual conference bridging or switching takes place in an emitter-coupled-logic signal processing computer, the Lincoln Laboratory Digital Voice Terminal (LDVT). A peripheral A/D-D/A system along with multiplexing-demultiplexing allows for twenty users to dial in for conferencing, as well as connections for three external voice coders. With this arrangement, conference bridging/switching is under software control of the LDVT, and can be changed dynamically, if necessary, to explore various conference connections. In addition to the switchboard function, the signal processing machine in conjunction with a large storage memory can also provide up to two satellite hop delays for each of the twenty users, or more delay for fewer users. This feature adds the final variable parameter needed for all conference experiments. A connection between the touch-tone modem and the PDP 11/45 by way of a special interface keeps the 11/45 up to date on conferees and queues. A second interface between the signal processing machine (switchboard) and the 11/45 allows for energy and silence measures to be sent to the 11/45 for statistics gathering, also indicates who talkers and interrupters are, and finally provides a path for the LDVT signal processing machine to be loaded from the 11/45.

b. Human Factors Effort

Given a conferencing facility, it is relatively simple to configure a conference structure, have people use it, and then elicit subjective statements about its worth, quality, ease of use, and so on. However, it is more difficult to quantify the performance of a particular conference configuration by using trained subjects as conferees, performing a standard task with predictable solutions while examining in detail the statistical track of the conference, and recording the conference for examination upon replay. If, in addition, one elicits subjective opinions from these subjects, they may have more merit because of the many variables which have been eliminated. In this regard, Lincoln contracted with BBN for human factors consultation, including experiment design and data evaluation, in order to pursue the more subjective work statement questions. The tasks which have been performed to allow for proper human factors evaluation of various conference experiments are:

(1) Collect and train subjects,

(2) Develop suitable scenarios and problems that allow for rich interaction between conferees, yet provide quantifiable information,

(3) Schedule subjects, hardware, and software for a conference experiment, and

(4) Observe the conference in progress and collect dynamic data pertinent to the human factors evaluation.

3. Summary of Results

Following is a summary of our studies, analyses, and experiments to date:

(a) For analog or wideband PCM, a full-duplex analog bridge is more satisfactory than any switching technique.

(b) For narrowband encoding (LPC or channel vocoders) there is no known bridging technique which can give satisfactory results.

(1) Conversion to analog followed by summation results in unsatisfactory tandem vocoding when only a single speaker is active and even worse performance when more than one speaker is active simultaneously, since such vocoders are theoretically incapable of reproducing a mixture of two or more voices.

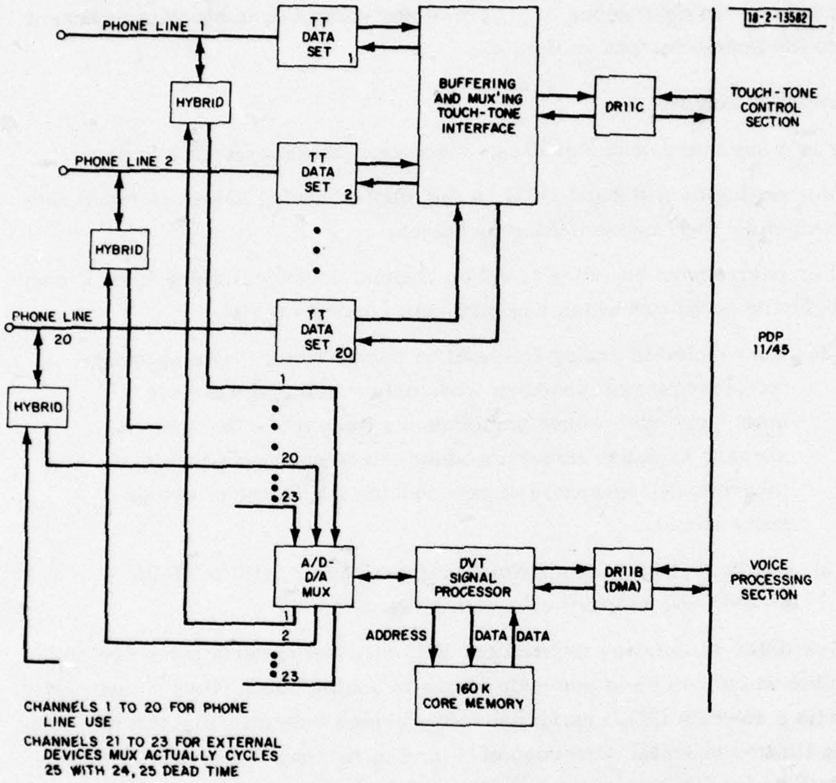
(2) Rapid switching on a frame-by-frame basis fails to yield an effective mixture of vocoded voices.

(c) For delta-modulation techniques, majority-voting techniques are possible which can yield approximations to analog summation. Experience with a 16-kbps CVSD majority-voting bridge indicates that the technique is limited to small conferences (3 or 4 participants) because of degradation of the signal-to-noise ratio which becomes worse as conference size increases.

(d) The lack of any effective bridging technique for narrowband encoding mandates the use of signal-selection techniques for conferences which must accommodate narrowband users.

(1) Experimental results show that while conference participants prefer bridging to selection, the effectiveness of the conference – as measured by problem solution time – is not reduced by the use of selection techniques.

(2) For the signal-selection techniques tested there were no significant interactions between the speech-encoding technique and the conferencing technique.



CHANNELS 1 TO 20 FOR PHONE LINE USE

CHANNELS 21 TO 23 FOR EXTERNAL DEVICES MUX ACTUALLY CYCLES 25 WITH 24, 25 DEAD TIME

Fig. II-1. Conferencing system.

(e) For all techniques, bridging as well as signal selection, increasing conference size and/or transmission delay makes conferencing more difficult. No interactions were observed between these variables and the conferencing technique.

(f) Our recommended "best" conference system would use a simple broadcast, voice-controlled technique with an optional "interrupter" channel to allow the selected speaker to hear a would-be interrupter. The "interrupter" channel is of doubtful value for small informal conferences, but is expected to be useful to the chairman in a large formal conference.

B. CONFERENCING FACILITY

1. Conferencing Hardware

The Lincoln Laboratory Secure Voice Conferencing Facility was described in detail in Ref. 1. All the system features described in Appendix A of that report have been constructed and operate as planned. For completeness, the appendix is repeated here with minor changes.

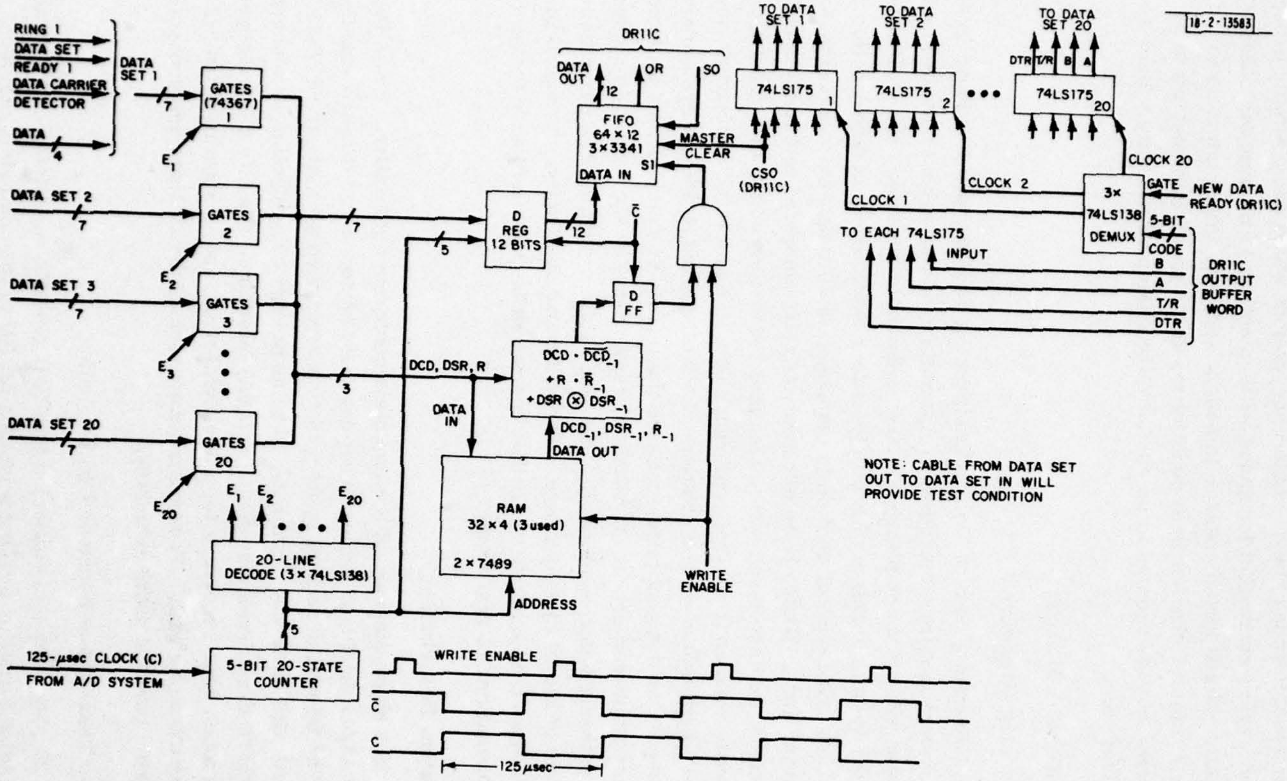
Basically, the system consists of two independent sections – a control section and an audio conditioning section. The control section is composed of 20 touch-tone data sets connected to dial-up Bell System lines. These lines are automatically answered to establish a user-to-computer connection, and are then used to transmit touch-tone commands from a user to a PDP 11/45. These commands control conference configurations and conference queues in real time. The audio conditioning section consists of a multiplexed A/D-D/A system and a large buffer memory connected to a signal processing machine (LDVT) which allows audio connections to be made arbitrarily between users. In addition, three ports on the A/D-D/A system are to be used to connect external voice equipment. The large buffer memory can implement delays of up to 0.5 sec for each of the 20 dial-up users. For additional flexibility, the signal processor is also connected to the 11/45 so that the control inputs can be used to modify the switching and signal processing operations in real time.

a. System Description

Figure II-1 is a block diagram of the complete conferencing facility. From the point of view of the PDP 11/45 machine, two external devices are connected through standard DEC interface circuits. The telephone control system is connected through a standard DR11C single-word interchange board with interrupt capability. The audio-switching section is connected through a more flexible DR11B direct-memory-access (DMA) interface. Twenty 2-wire phone lines are connected to the touch-tone receivers for the control path, and to a set of hybrid (2- to 4-wire) transformers for the audio path. Four wires from each of the 20 lines are connected to an A/D-D/A converter port for audio switching.

b. The Touch-Tone Receiver Control Path

Each of the 20 phone lines is connected to a Bell type 403 tone data set which automatically responds to a ringing signal by passing a ringing bit (R) to the computer interface. If the computer raises a data terminal ready (DTR) bit, the data set will answer the line and set up to receive control tones by transmitting a data set ready (DSR) bit. When a user presses a tone

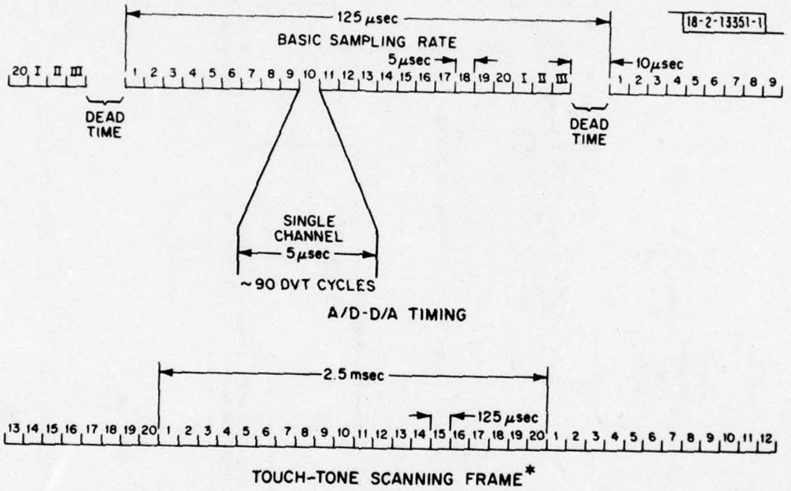


NOTE: CABLE FROM DATA SET OUT TO DATA SET IN WILL PROVIDE TEST CONDITION

Fig. II-2. Touch-tone interface, 20 data sets to DR11C.

button, the data set will signal the computer with a data carrier detector (DCD) bit, and a 4-bit tone code. The computer can listen for these tones, have the data set transmit three single-frequency responses, or hang up.

Figure II-2 presents the interface between 20 data sets and the DR11C. The basic interface function scans the 20 data sets for activity by comparing a new status word from each channel with a previous stored status word from the same channel. Each previous channel status word has been stored in the 32- × 4-bit RAM. Only the three status bits (R, DCD, and DSR) need be stored for comparison against the latest word. If there is a change in any of these bits where change is defined as: $\mathrm{DCD} \cdot \overline{\mathrm{DCD}}_{-1} + \mathrm{R} \cdot \overline{\mathrm{R}}_{-1} + \mathrm{DSR} \otimes \mathrm{DSR}_{-1}$, then the present word, including a 5-bit code for channel identification, is clocked into a first-in/first-out (FIFO) buffer and an output request is set. The 20 data sets are scanned in a cycle of 20 of the 8-kHz (125-μsec) samples (see Fig. II-3), so that a complete scan requires 20 × 125 × μsec = 2.5 msec. Each data set is controlled from the interface by a 4-bit register which is loaded under program control from the PDP 11/45 – DR11C path.



* NOTE: THE PDP 11/45 DOES NOT SEE THIS TIMING. IT COMMUNICATES VIA INTERRUPTS FROM THE TOUCH-TONE INTERFACE.

Fig. II-3. Conferencing system timing.

c. The Audio Conditioning Section

As Fig. II-1 indicates, the audio conditioning section consists of three subsections: an LDVT signal processing computer, a multiplexed A/D-D/A system which is controlled by and communicates with the LDVT, and finally a large (160K) core memory which is controlled by the LDVT. The LDVT, in turn, can also communicate with the PDP 11/45 through a DMA interface called a DR11B.

d. The Multiplexed A/D-D/A System

The A/D-D/A system is shown in Fig. II-4. It is connected to the channel 0 input and output ports of the LDVT and consists of an A/D section, a D/A section, and some multiplexing timing registers.

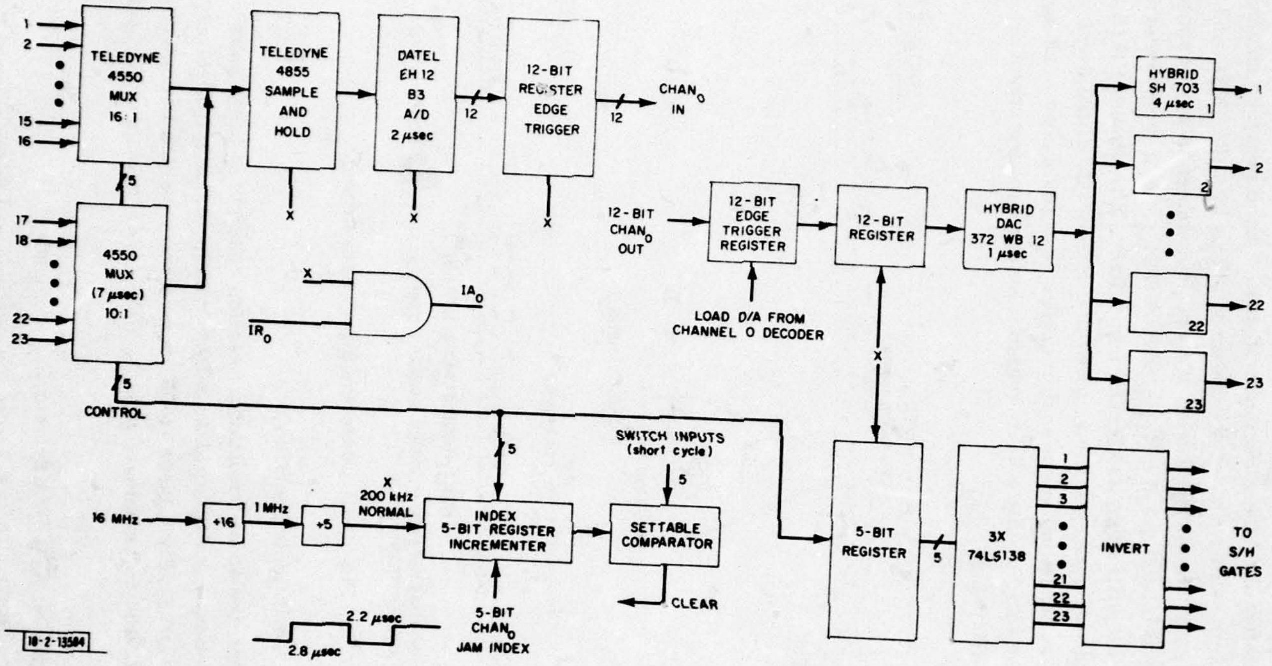


Fig. II-4. A/D-D/A-MUX system.

The A/D section can accept up to 32 input analog signals multiplexed through two Teledyne 16:1 gates (only 23 inputs are used). These multiplexer gates drive a sample-and-hold (S/H) gate which drives, in turn, a 12-bit A/D converter. The multiplexed input is controlled from a 5-bit register incrementer which can be loaded with a 5-bit word asynchronously so that random access conversion of any input channel can take place; or, a standard input clock will increment the register by one during each cycle and clear at some settable value. In other words, the input multiplexer can be stepped randomly, or cycled through a fixed pattern. A normal input conversion rate is 200 ksec/sec, although an external clock can be used. The input A/D 12-bit word is read on input channel 0 of the LDVT, either as a forced input or an interrupt.

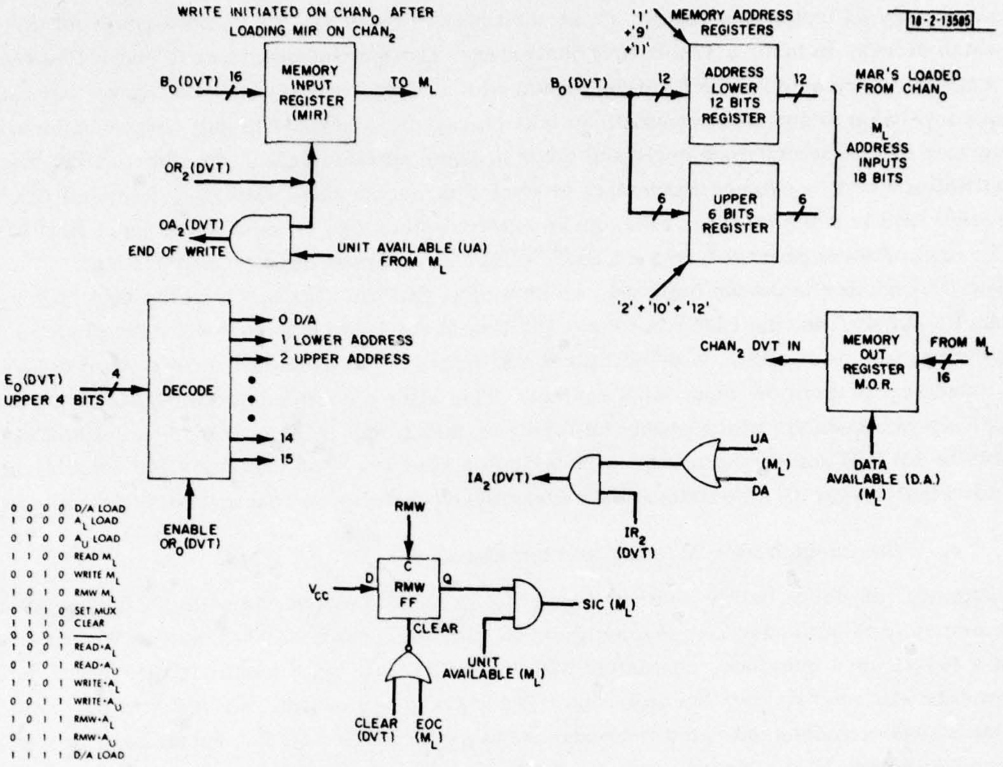
The D/A section is double buffered, which means that the user can load the D/A buffer on a channel 0 output from the LDVT but the transition of the D/A converter will take place on the next synchronous clock edge. A demultiplexer S/H gate is controlled by a 5-bit word delayed by one clock cycle from the input MUX control. This allows for the delay in D/A conversion. The D/A section consists of the double buffering, a fast 12-bit D/A converter, a set of 23 (expandable to 32) S/H gates, and a 5-bit decoder pulse steerer. The choice of S/H outputs rather than individual slower D/A registers and converters was based on cost and wiring complexity.

e. The Large Buffer Memory and Interface

Basically, the large buffer memory is a 128K by 20-bit core memory plus a 32K by 20-bit core memory, and both have a read-modify-write time of approximately 2 μsec. We have designed a 16-bit word interface, consistent with the LDVT data word length, although our delay experiments will require only 12-bit words. The input to and output from the memory (write and read words) are communicated from and to channel 2 of the LDVT. Actual read, write, read-modify-write, load address, and various hybrid commands to the large memory are transmitted from output channel 0 of the LDVT. Since this channel was designed as a 12-bit output to a D/A converter, 4 more bits are available to be decoded and used to steer data to places other than the D/A converter. The lower-left portion of Fig. II-5, the memory interface and channel 0 decoder, shows the decoding table. An output on channel 0 from the LDVT, with 4 upper bits zero or all 1, produces a standard D/A load. The other commands load upper and lower portions of the 18-bit address register, and start read, write, or read-modify-write cycles. Since the output on channel 0 is a 12-bit word, the loading of the address register is a two-command operation. The lower address (A_L) is 12 bits and the upper portion (A_U) is 6 bits. Presumably, only the lower register would be loaded for many applications requiring only one command. It is also possible to combine the address load with a read, write, or read-modify-write command. Two remaining commands set up the multiplex word and do a master clear.

f. The LDVT as Controller

The LDVT has a limited in-out system which has been modified to control the multiplexing system and the large memory. The present 4 channels of input and output are assigned as follows. Channel 0 outputs to the D/A converter, sets the MUX index, or controls the large memory. Channel 0 input receives data from the A/D converter. Channel 1 communicates with the PDP 11/45 through the DR11B interface. Channel 2 reads from and writes to the large memory (M_L). Finally, channel 3 remains as the link to M_X, the internal LDVT bulk memory. The 55-nsec cycle time of the LDVT allows for approximately 90 machine cycles during each 5-μsec A/D conversion cycle.



0	0	0	0	D/A LOAD
1	0	0	0	A_L LOAD
0	1	0	0	A_U LOAD
1	1	0	0	READ M_L
0	0	1	0	WRITE M_L
1	0	1	0	RMW M_L
0	1	1	0	SET MUX
1	1	1	0	CLEAR
0	0	0	1	———
1	0	0	1	READ·A_L
0	1	0	1	READ·A_U
1	1	0	1	WRITE·A_L
0	0	1	1	WRITE·A_U
1	0	1	1	RMW·A_L
0	1	1	1	RMW·A_U
1	1	1	1	D/A LOAD

Fig. II-5. Large memory interface (M_L) channel 0 decoder.

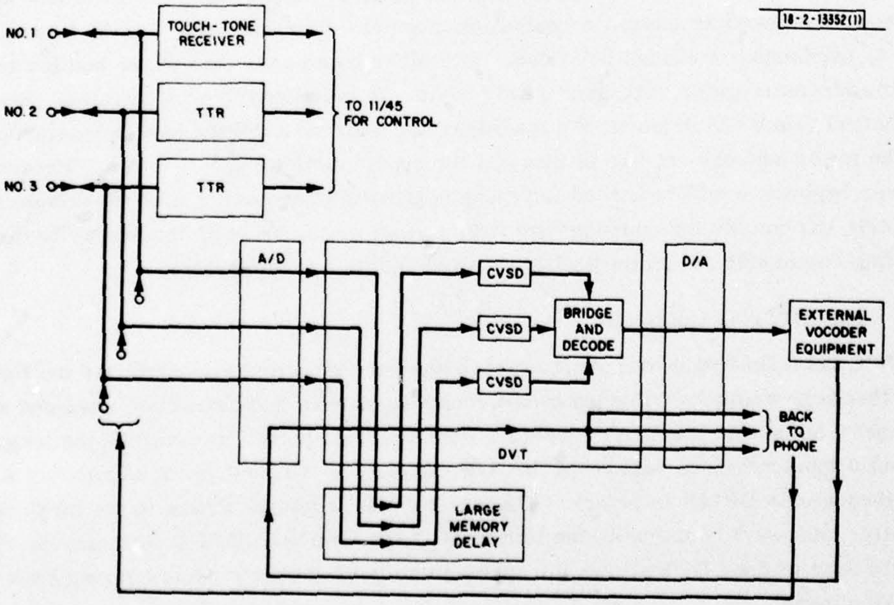


Fig. II-6. Conference example.

2. Conference Experiment Examples

Figure II-6 shows the conferencing facility as it might be configured for a 3-party conference. This example shows a conferernce which is bridged at the delta modulated bit level, tandemed in a narrowband vocoder, and then distributed to the conferees.

The three participants form the conference by dialing up one of the 20 phone numbers, and communicating via touch-tone to the PDP 11/45 conference control program. The LDVT software is loaded via the 11/45 to implement CVSD encoders for each of the participants, effect the bit stream bridging, delay the audio inputs by fixed or time-variable amounts, output the decoded bridged signal to an externally connected vocoder (on channel 21, 22, or 23), and receive the output of the vocoder tandem back on the corresponding A/D channel for distribution to all the conferees, or all except the one talking.

If a fourth person wishes to join the conference, he calls in and interacts with the control software scanning the touch-tone interface. Then, flags are activated in the LDVT to enable another A/D-D/A channel and include the fourth stream in the bridging and distribution.

Figure II-7 indicates the physical layout of conferencing equipment aside from the PDP 11/45, and the large core memory used for delay.

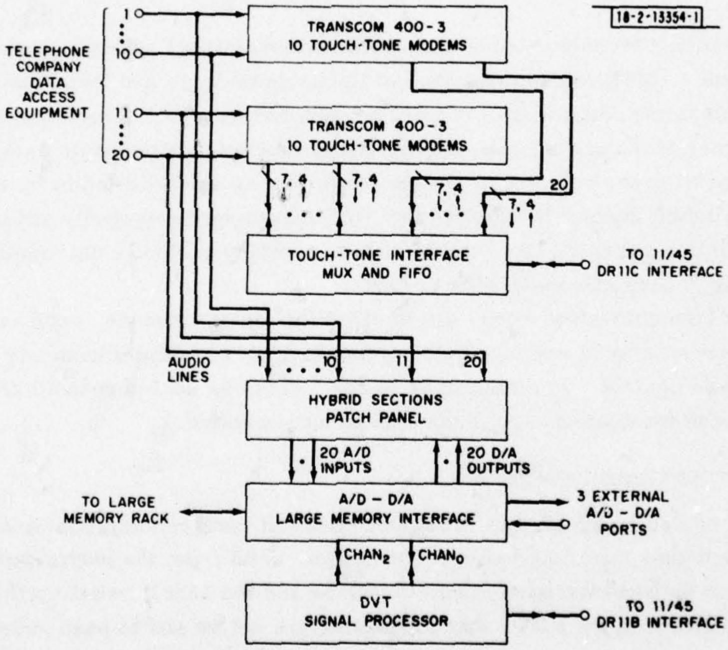

18-2-13354-1
TELEPHONE COMPANY DATA ACCESS EQUIPMENT
1
10
11
20
TRANSCOM 400-3 10 TOUCH-TONE MODEMS
TRANSCOM 400-3 10 TOUCH-TONE MODEMS
7, 4
7, 4
7, 4
7, 4
20
TOUCH-TONE INTERFACE MUX AND FIFO
TO 11/45 DR11C INTERFACE
AUDIO LINES
1
10
11
20
HYBRID SECTIONS PATCH PANEL
20 A/D INPUTS
20 D/A OUTPUTS
TO LARGE MEMORY RACK
A/D – D/A LARGE MEMORY INTERFACE
3 EXTERNAL A/D – D/A PORTS
CHAN2
CHAN0
DVT SIGNAL PROCESSOR
TO 11/45 DR11B INTERFACE

Fig. II-7. Conferencing rack.

As mentioned earlier, statistics about activity, coincidence of talkers, etc. can be gathered on-line by way of the LDVT link to the 11/45.

3. Conferencing Software

In this section, the software which is common to all simulations involving the hardware conferencing facility is discussed. The simulation of a particular conferencing technique is realized by extending this common software base to effect the desired bridging or switching technique. The commonality follows from a decision to fix the information format exchanged between the LDVT signal processor and the PDP 11 control and data collection processor. As a result, all voice energy switched and bridged conferencing simulations can use the same PDP 11 programs for control, data collection, and data reduction. However, the PDP 11 code must be specialized for the conference technique which uses touch-tone signaling.

a. PDP 11 – LDVT Communications

Communication between the LDVT and the PDP 11 involves the transfer of blocks of twenty 16-bit words every 20 msec. There is one word in each block for each possible conference participant. A bit in each word indicates to the LDVT whether the corresponding phone is to be considered active or not. If a phone is marked as active, the LDVT program will treat the signal from that phone according to the conferencing algorithm in effect. In addition, the program will look for speech activity from that phone by accumulating the sum of the absolute values of the PCM readings for each 2-msec interval. If the sum exceeds a threshold during any of the ten 2-msec intervals in a 20-msec reporting period, the LDVT will indicate that fact to the PDP 11 program by setting a speech activity bit in the corresponding word of the block sent to the PDP 11.

If the conferencing technique being simulated involves signal selection based on voice energy detection, the LDVT program carries out the decision logic and indicates its decision by setting another bit in the communication word corresponding to the selected speaker. If a speaker/interrupter technique is being simulated, yet another bit is set to mark the interrupter. The 20-msec reporting period determines the resolution at which switching times are known, but the actual switching instant is quantized by the 2-msec speech-activity accumulation time. The loss in resolution resulting from the 20-msec reporting period is not significant since speaker switching occurs at a much slower rate.

The block of communication words can be used for other purposes, such as allowing timing and amplitude threshold to be communicated from the PDP 11 console keyboard to the LDVT which lacks console control. In simulations to date, only one such threshold value is used. Its meaning varies with the conferencing technique being simulated.

b. PDP 11 Control Program

The PDP 11 control program has two functions in all conferencing simulations. One is to command the touch-tone interface hardware to answer calls from the participants and thus effect the connection between the participants'[2] phones and the LDVT switching/bridging processor. The other is to indicate to the LDVT that the phones are active and to pass run-time parameters to the LDVT program.

The control program can be given commands from the console keyboard to indicate which phones are to be answered and how many participants are to be accepted in the conference. While it is possible for n active phones to be distributed arbitrarily over the available phone numbers, current software limits a conference of n participants to the first n phones in the order of their connection to the conferencing hardware.

Two versions of the control program are available. In the first (the most commonly used), the commands to answer the first n phones are issued prior to any participant dialing activity. In the second, the control program waits for a ringing signal and issues the command to answer when the ringing signal is observed and the phone is one of the n to be accepted. In the first case, all conference participants hear the tone generated by the answering hardware and are made aware that someone is entering the conference. In the second case, the tone is inhibited because the control program does not tell the LDVT that the new phone is active until the end of the answering tone. Our human factors research has not yet addressed the question as to whether participants should be advised of the arrival and departure of other participants.

c. LDVT Switching/Bridging Program

The LDVT program receives PCM inputs and provides outputs for all phones connected to the conferencing facility. The A/D-D/A multiplexer scans through the 20 phone lines and three speech encoder ports, allowing 5 µsec per line for processing in the LDVT. This time allows approximately 90 instructions to be executed in the LDVT for each phone line. These instructions must provide for the execution of the basic signal selection or bridging algorithm required for the conferencing technique being simulated, as well as to allow for speech-activity detection and, in some cases, delays corresponding to satellite transmissions. In addition, small delays are introduced to improve the operation of the speech-activity detectors in voice-switched signal selection conferences by allowing the detector to anticipate threshold crossings. The delays are realized by storing the PCM speech samples in a large core memory attached to the LDVT. When satellite delays as well as anticipatory delays are used, almost all of the possible 90 instruction executions are needed. Very careful coding is required to avoid exceeding the 5-µsec timing constraint.

The exchange of information with the PDP 11 is handled in the 10 µsec which remain in the basic 125-µsec frame (8-kHz sampling rate) after servicing the 20 phone lines and three speech encoder ports. Word transfers take place on 40 (20 in each direction) of the 160 frames which occur during the 20-msec reporting period. The transfers are spaced to allow the slower PDP 11 hardware and software to handle them without difficulty.

d. PDP 11 Data Collection Program

As discussed above, the LDVT sends 3 bits of information to the PDP 11 for each participant during every 20-msec reporting period. These bits tell whether or not the participant was exhibiting speech activity, was the selected speaker, or was the selected interrupter during the previous period. We call the combination of these 3 bits the "state" of the participant. The data collection program observes the state of each participant and makes up a disk file which has an entry for every change of state. The entry shows the new state, as well as a time marker equal to the number of 20-msec report periods since the start of the conference.

Data collection begins when the conferencing simulation program starts, and ends when the program is manually stopped. To allow experimenters to mark off time periods of interest during a conference, a push button is available which when pushed introduces a signal into an otherwise unused phone channel. The signal is noted in the collected data. A companion button can be pushed to add an audible tone to the audio recording normally made during a conferencing experiment. This tone can be used to correlate the marked point in the data with the conference content.

e. Data Reduction Software

To aid in the analysis of conferencing experiments, a data reduction program has been developed which produces both global summary information and/or a detailed step-by-step history of the conference interactions. The data reduction program operates on the files established by the data collection program.

The global summary information is produced in the form of several charts:

(1) For each speaker, a count of the number of times a transition was made into each of the possible states.

(2) For all the speakers combined, a histogram of the durations in the various states.

(3) For each speaker, the total time spent in each state.

(4) For each speaker, the total time spent speaking, i.e., the sum of the speaker's talk spurts. A talk spurt is defined as a "smoothed" time interval when the speaker's energy level was above threshold. To provide the smoothing, "small" silence gaps (i.e., intervals in which energy is below threshold) are considered as part of the talk spurts. After these "silence gaps" are filled, any resulting talk spurts that are suitably small are considered irrelevant noise and are disregarded in the final tabulation. The two constants, the size of the silent gaps and the size of the ignored spurts, are easily modified. This method of tabulating talk spurts closely simulates the perceptions by humans who normally consider a talk spurt as a substantial interval between major silences, ignoring small silences between syllables or words.

(5) For each speaker, a histogram of the number of talk spurts of various durations. Talk spurts are as defined in the previous paragraph.

(6) For the conference as a whole, the total times n phones were simultaneously over threshold. This provides a convenient measure of the amount of talk as well as conflict (simultaneous talk) in the conference. It should be noted that the feature measured here is simply energy level above threshold rather than talk spurts. An example of summary outputs from a conference experiment is shown in Appendix B.

A detailed step-by-step picture of the conference is provided by an audit trail output. The time axis extends horizontally and speakers are plotted in the vertical axis, analogous to a strip-chart recording. At each intersection of time and speaker, an indication of the state of that speaker for that time interval is presented.

Duration of time interval is selectable when the audit trail program is run. Two distinct audit trails are available based on the selection of time interval for each tick mark in the time axis.

If each tick mark is selected to be one 20-msec period, then the audit trail shows each actual transition as it occurs. No merging need be done, since 20 msec is the basic time ' nit for indicating transitions to the PDP 11.

If the tick mark is more than one 20-msec period, then the audit trail shows merged information. For example, during a 1-sec interval (fifty 20-msec periods) a given speaker may have been both the designated interrupter and the designated speaker. The first figure of Appendix B is an audit trail with a 1-sec marking interval.

In addition to providing global summary information and step-by-step pictures, the analysis is useful as an aid to debugging and fine-tuning the conferencing algorithms.

C. CONFERENCING TECHNIQUE SIMULATION PROGRAMS

In this section, we describe the programs which implement the simulations of conferencing techniques which have been subjected to experimental evaluation. Most of these programs are capable of simulating a range of systems by varying parameters which are either run-time inputs to the program or require reassembly. All the programs make use of the basic software discussed in Sec. B-3 above for data collection and control. All except the CVSD majority voting bridge can be run with or without simulated transmission delays. The presently available programs simulate equal delays for all participants if the delay option is requested.

The conferencing techniques which have been evaluated fall into two broad categories – bridging techniques and signal-selection techniques. In the first category, each participant hears some combination of the speech signals produced by the other participants; in the second category, a participant hears the speech of only one of the other participants at a time.

1. Bridging Techniques

a. Analog Bridge

The basic bridging technique is a summation of the signals from each of the participants. In ordinary telephone conference calls, this summation is accomplished by analog addition of the signals from the participating phones. In our simulation, the addition is performed digitally using the 12-bit PCM samples from the A/D multiplexer. The 12-bit values are accumulated into a 16-bit word allowing the sum to be as large as 16 times the maximum signal from any individual phone without causing overflow. As a result, the probability of overflow of the sum is negligible under any normal conferencing situation. However, the sum may exceed the 12-bit range of the output D/A equipment if two or more speakers are producing very loud speech sounds at the same time. The bridge program checks for output values which would exceed the range, and limits the actual outputs to full-scale values.

In the event that transmission delays are to be simulated, it is important that participants not hear their own voices delayed by more than a few tens of milliseconds. To avoid problems with long delays, the analog bridge program subtracts each participant's input from the output returned to that participant. The effect is the same as would be achieved by providing n distinct summations, each summing n – 1 participants' signals. The technique of subtracting out the input works perfectly for a digital summation so long as the internal sum does not overflow the range of the accumulator.

Even though the participant's own voice is not transmitted directly to his receiver, in the event that the delay option is in effect he will hear his own delayed voice at a low level (approximately 30 dB down from the other talker) due to the imperfect action of the hybrid circuits which interface the conferencing facility to the telephone system. It is our observation that this delayed signal, while readily detectable if one listens for it, is not an important disturbing factor in an actual conferencing situation where the participants are concentrating on the scenario contents.

b. CVSD Majority Voting Bridge

For delta modulation encoding techniques, it is possible to approximate the action of an analog bridge without decoding the signals to be summed. For example, in CVSD encoding, 2-bit sequences may be interpreted as follows:

00	slope is consistently negative
01	slope changes from negative to positive
10	slope changes from positive to negative
11	slope is consistently positive

In our majority voting bridge, the most recent 2 bits from each input encoder are examined and votes are indicated as follows:

00	cast a vote for a negative output slope
01	cast no vote (abstain)
10	
11	cast a vote for a positive output slope

If the majority of input encoders indicate votes for a negative output slope, an output of "0" is generated. If the majority vote is positive, an output of "1" is generated. If a tie vote is registered or all inputs abstain, then the output is set to the complement of the previous output.

The output of such a majority voting bridge exhibits a signal-to-noise ratio (SNR) which becomes progressively worse as the number of inputs increases. The noise increases because the voting process gives equal weight to all input slope information without regard to the magnitude of such changes. We feel that the noise increase limits use of the technique in its pure form to small conferences with, at most, three or four participants.

In order to increase the utility of the majority voting technique and extend it to larger conferences, we have added speech-activity detection to the bridge so that only those phone lines on which activity is detected are considered in the voting procedure. As a result, since most of the time in a conference only one participant is speaking, the speech quality will most of the time be no worse than one would expect from CVSD encoding. Only when two or more people speak at the same time (the order of 5 percent of the total speech time in our experiments) is there any degradation of the SNR due to the majority voting operation.

In our implementation the CVSD analysis, the majority voting, and output synthesis are all handled by the LDVT switching/bridging processor. Because of the heavy computing load associated with the CVSD analysis of the input signals, the simulation is limited to eight participants and the transmission delay option is not available.

In order to achieve 16-kbps CVSD speech encoding with the conferencing A/D multiplexer which runs at an 8-kHz rate, it is necessary to estimate every other sample by means of linear interpolation. This technique introduces a negligible error when the input speech is band-limited correctly for the 8-kHz sampling rate, as it should be for 16-kbps CVSD encoding.

Unlike the analog bridge simulation, the CVSD majority voting bridge does not subtract out a speaker's voice from the signal he or she hears, because the LDVT cannot handle the computations required to produce eight different outputs. Since this simulation does not include delay effects, the speaker hears this as normal sidetone.

2. Signal-Selection Techniques

Signal-selection techniques may be divided into three broad classes according to the means used by the participants to indicate their desire to speak. We use the term "voice control switching" (or VCS) to refer to techniques which sense the presence or absence of speech energy as input to the selection process. We use the term "push-to-talk" (or PTT) to refer to techniques in which a participant pushes and holds closed a button or switch while speaking. The switch action is presumed to be communicated to the conference controller by means of an extra order wire, by having the controller sense the presence of a carrier frequency, or any other method which can convey the state of the switch to the controller. Finally, we use the term "control signal switching" (or CSS) to refer to techniques which use touch-tone keys or the like to send control signals to the conference controller. The controller can, in turn, signal the participants by lighting indicator lights, using audible tones, or prerecorded speech messages to indicate that it is now time for the participant to speak. CSS differs from PTT in that the keys need not be held down while talking and the key action can precede the opportunity to speak and will be remembered by the controller.

Signal-selection techniques can also be differentiated according to the following dimensions:

(a) Full- or half-duplex. Can a participant hear the conference while he or she is speaking?

(b) Interruptible. Can a participant be interrupted before he or she has finished speaking?

(c) Time limitation. Does the controller put a time limit on how long a speaker can talk?

(d) Priority. Are there priority rules in effect to resolve conflicts or control interruptibility?

(e) Urgency. Can a participant with an urgent message indicate that fact to the controller and gain faster access to the conference "floor"?

(f) Speaker/Interrupter. Can a speaker hear a would-be interrupter without being interrupted himself?

(g) Chairman control. Does a person playing the role of chairman in a formal conference have any special control over the conferencing hardware?

Picking options from the above list and the three control classes can lead to a very large number of distinct conferencing capabilities to explore by simulation. We have chosen to start with relatively simple configurations. To date four signal-selection simulation programs, described below, have been developed for formal experimentation.

a. VCS Simple Broadcast – Noninterruptible

In this simulation, a speaker is selected and his or her speech is broadcast to all other participants. The selected speaker hears nothing while selected; he or she is selected by noting the first participant who exceeds the speech-activity energy threshold after a period of silence. If more than one participant is found above threshold in the 2-msec decision interval, the one with the largest energy will be selected. In the unlikely event of a tie, the decision is resolved using an implicit priority determined by the order in which the program scans the phone lines.

Once selected, a speaker remains selected until his speech energy has fallen below threshold and remains below threshold for a time interval which is a run-time parameter for the simulation. The currently favored value for the parameter is 400 msec. With such a value, the speaker can retain the "floor" as long as desired by continuing to speak without pausing other than very briefly. The silence duration which allows an interrupter to gain the floor is a compromise value. A duration longer than about 400 msec allows the speaker to retain the floor more easily against a would-be interrupter, but it tends to cause clipping of the start of the next speaker's utterance.

b. VCS Speaker/Interrupter

This simulation is very much like the VCS Simple Broadcast – Noninterruptible simulation described above. The rules for selecting and retaining a speaker are identical. The difference lies in that, while a speaker is selected, if some other participant starts to talk his or her speech is fed to the originally selected speaker as a so-called "interrupter." The interrupter continues to hear the speaker. The interrupter's speech is not heard by the other participants unless the selected speaker stops talking. The rules for selecting and retaining the interrupter are the same as those for the speaker, except that the speaker's phone line is not scanned while searching for an interrupter.

It may be the case that an interrupter will succeed as speaker when the present speaker finishes, but there is nothing in the program to guarantee such a succession. The selection of a new speaker proceeds without knowledge of the interrupter and is based solely on the first or loudest participant in the 2-msec interval in which a decision is made.

If only two participants are active in a conference for a period of time, the speaker/interrupter technique gives them a full-duplex communication capability. Other participants acting as listeners will miss some of the resulting dialogue to the extent that the two talkers speak simultaneously.

c. VCS Simple Broadcast – Interruptible

In this simulation, the selection of a speaker to be broadcast is made at nearly every 2-msec speech-activity decision interval. The role of speaker goes to the participant with the highest energy value above threshold. However, once a speaker has been selected, he or she is allowed to continue as speaker for a period of time which is a run-time parameter of the program. Typical values for the parameters are 500 to 600 msec.

This conferencing technique allows one participant to attempt to interrupt another at any time by simply starting to talk. The louder the interrupter speaks, the more likely will the interruption succeed. Of course, a conference will degenerate into chaos if the participants keep trying to interrupt each other. In practice, they do not do so except for brief periods.

The motivation for an interruptible broadcast signal selection technique is to try to approach the capability for interruption found in the analog bridge. In that case, it is often possible to follow the content of both talkers' utterances as well as to recognize the identity of the interrupter. The value of 500 to 600 msec was chosen for the hang-on time after a speaker switch, to allow some chance of recognizing the identity of the interrupter while retaining some of the content of the speaker's utterance. Longer hang-on times give more advantage to the interrupter. Shorter times tend to cause the speech to become unintelligible without providing any useful information about the interrupter.

To avoid losing good speech unnecessarily because of the hang-on time, the hang-on state is aborted if the newly selected speaker's energy falls below threshold and remains below for 60 msec, such as might occur for a noise burst.

d. PTT Broadcast – Half-Duplex

To allow experimentation with PTT conferencing techniques, a number of Lincoln Laboratory telephones have been equipped with switches which can open and close the connection between the transmitter and the phone line. The switches are mounted on the handsets in such a way as to allow easy operation by the hand which holds the handset.

In order to simulate the full operation of a PTT conferencing technique, it is necessary for the LDVT program to be able to sense the position of the handset switch. Since no additional order-wire circuit is available, the program must depend upon the difference in observed noise on the line between the open- and closed-switch conditions. By subtracting a reference dialed-up channel from the input signal to reduce the importance of a 60 hum signal normally present in both conditions, it has been possible to achieve fairly reliable detection of switch position, particularly if there are some high-frequency components present in the room noise at the telephone site.

The PTT signal-selection program chooses as the broadcast speaker the first participant whose switch is observed to be closed. Having once been selected, the participant will continue to be the conference speaker until the switch is opened. At that time the program will seek another speaker. If two or more switches are closed at the same time, the program decides which participant is to be the speaker according to an implicit priority determined by the order in which the phone lines are scanned.

To simulate the effect of half-duplex communication links, the program shuts off output to all phones for which a closed switch is detected. As a result, a participant who closes his switch and speaks for a time cannot tell directly whether or not he has been heard by the other conferees. If he is the selected speaker, however, he can continue to talk without fear of interruption until he releases his switch.

D. HUMAN FACTORS

Included among the human factors activities undertaken by BBN in support of the Lincoln Laboratory Secure Voice Conferencing Test and Evaluation program were the following:

(1) Definition and development of scenarios suitable to the evaluation of voice conferencing system alternatives.

(2) Acquisition and training of subjects.

(3) Identification of appropriate experimental designs, data to be collected, and methods of analysis.

(4) Analysis of results.

Discussions of the essential elements of these activities appear below. More complete discussions of details of scenario criteria, experimental procedure, and data analysis can be found in BBN Report No. 3681 to be published in December 1977.

1. Definition and Development of Scenarios

a. Criteria for Selection

On the basis of a search of literature relating to group problem solving, human information processing, etc., a number of criteria for the definition and selection of teleconferencing scenarios were developed. As summarized earlier in the Semiannual Technical Summary,[1] these are as follows:

(1) A given problem scenario should be usable over the entire range of conference sizes to be evaluated, and its difficulty level should be independent of size. Furthermore, scenarios should be constructed in such a way that they can be reused with a given set of conference participants.

(2) Problems selected should be intrinsically interesting to subjects, and the testing situation should promote highly motivated performance.

(3) Scenarios employed should permit a variety of objective performance measures, including gross measures such as solution time and solution quality, and fine measures of communication and system effectiveness and dynamics, such as number of messages per speaker per unit time, average queue length and speaker waiting time, and duration of pauses between messages.

In addition to these general criteria, we considered it important to identify and develop scenarios that would place reasonably severe demands on the bandwidth available within each of the systems of interest, even if such an approach might lead to conferencing problems that at least superficially were dissimilar to those that might be encountered under operational conditions. This consideration arose as the result of an early expectation that, because of the generally high speech-transmission qualities of the telephone systems to be evaluated, all conferencing arrangements might prove to be equally satisfactory unless (and, possibly even if) tests were conducted on a "worst-case" basis.

Finally, we thought that an ideal scenario would be one whose requirements could be easily taught to experimental subjects who might differ in vocational specialty, level of formal education, intelligence, etc. Satisfaction of this criterion was expected to aid and abet accomplishment of a variety of practical goals such as reducing the difficulties associated with selection and replacement of subjects, and increasing the generality of results of teleconferencing experiments.

b. Scenarios Developed

Efforts at joint satisfaction of the above criteria led to formulation of four different types of scenario.

Two of these represent relatively pure tests of the speed and ease with which conferees can pass information around a conference. The paradigm employed is one in which the content of the current speaker's message uniquely cues a message in the possession of one of the listeners. When the speaker completes his input, that listener then becomes speaker and disseminates his message, cueing a third party, etc. In both tasks – one involving the transmission of digit sequences, the other the transmission of data concerning orientations of line segments superimposed on the cells of matrices – conferees are instructed to proceed as quickly as possible consistent with a low error rate.

The third scenario r^quires conferees to achieve an optimal allocation of resources in accord with specified constraints. In this scenario, each conference member is provided with information concerning the "home" location, "work" location, and "desired arrival time" of one or more fictitious commuters. He is also given a map showing the locations of towns identified with the problem, and a listing of possible car pools that might be formed between his commuter(s) and those assigned to other members of the conference. An experimental session begins with conferees exchanging information about locations and times, and proceeds to an interactive problem-solving phase in which conferees attempt to generate an optimal pooling and routing of commuters.

The car-pool task has proved to be an effective medium for the generation of relatively unconstrained dialog, and has been utilized almost exclusively in experimentation to date. An example of a problem actually employed during one of the sessions is contained in Ref. 1.

The last of the four tasks developed during this phase requires conference members to reinstate the sequence of sentences in corpora of text selected from newspapers, magazines, and books and randomized by the experimenter prior to the session. Although this task has also proved useful as a dialog generator, the fact that several "incorrect" orders are often at least as compelling as the "correct" order makes for considerable difficulty in scoring. As a result, the task has not been used since a better alternative – car pool – was developed.

2. Acquisition and Training of Subjects

a. Subject Population

In our judgment, the sequential development of hardware and software support capabilities within the conferencing testbed and the exploratory nature of the research program argued strongly for a stable, well-trained, and more-or-less continually available subject population. With the expectation that conferences containing 12 to 14 conferees could be supported by the end of the first year's effort, and that normal processes of attrition would reduce the subject population by at least one-third over the period, we concluded that 22 to 24 subjects, each of whom would serve for 50 experimental hours, would be required. From among approximately 27 volunteers from the Laboratory, 22 subjects were finally chosen in such a way as to secure the best obtainable ratios of females to males (15:7), and professional to clerical staff (9:13) available. As a result of attrition, these ratios now stand at (9:6) and (8:7), respectively.

b. Training Procedures

For training purposes, the population of subjects was divided into three subgroups of 4 persons each and two subgroups of 5 persons each. All groups received 5 hr of training prior to entry into the first experimental session. A summary of activities pursued during each of the hour-long sessions appears below:

Hour 1: Description by experimenter of a number transmission task; completion of three practice problems under face-to-face conditions; completion of two practice problems under analog bridge conference conditions.

Hour 2: Description by experimenter of a line-matrix task; completion of two practice problems under face-to-face conditions; completion of two practice problems under analog bridge conference conditions.

Hour 3: Description by experimenter of car-pool problem; completion of one practice problem under face-to-face conditions.

Hour 4: Completion of one practice problem under face-to-face conditions; completion of one practice problem under analog bridge conference condition.

Hour 5: Completion of two practice problems under analog bridge conditions.

Our primary goals during this relatively lengthy training period were (1) to assure that the subjects were thoroughly acquainted with the requirements of each task, and (2) to afford subjects ample opportunity to develop successful conceptual strategies for solution of the various problem types. Satisfaction of this latter goal was important, we felt, for assuring that differences in conference performance which might later be encountered during the evaluation of alternative teleconferencing systems were attributable to differences in ease and efficiency of communication, rather than to random variations associated with continual search for problem-solving methodologies.

3. Experimental Measures, Conditions, and Procedures

a. Summary of Experimental Measures

Because this research has been exploratory in nature, an effort has been made to define as wide a variety of conference measures as possible. The measures of current interest fall into three general categories:

Category 1: Gross measures of total conference performance, such as time required to complete an assigned task, quality of task solution, number of alternative solutions proposed.

Category 2: Fine measures of performance of individual members and of the conference as a whole, such as total time spent by each member speaking, number of times each speaker was interrupted, frequency with which 2, 3, . . . , n members were attempting to speak simultaneously, average length of talk spurts, ratio of total speaking to total quiet time during session, average queue length.

Category 3: Attitudes and opinions of subjects regarding the relative ease or difficulty of using the various teleconferencing systems in the course of problem solution, recognizability of speakers on the basis of the sounds of their voices, perceived ease or difficulty of interrupting another speaker when one has something to say.

b. Summary of Experimental Conditions

Experimental efforts during this stage of evaluation have been directed at two primary goals: (1) determination of the extent to which conference performance may vary as a function of conference size; and (2) initial estimation of the relative ease of use of teleconferencing systems available within the testbed. These goals, pursued concurrently with the development and refinement of techniques for performance assessment, have been at least partially satisfied for conferences of 4, 8, and 12 persons using an analog bridge, and for conferences of 8 persons using speaker/interrupter and simple broadcast VCS with delay. A complete listing of all experiments completed to date appears in Table II-1 below.

TABLE II-1

LISTING OF FY 77 TELECONFERENCING EXPERIMENTS

T/C Conditions Compared	Number of Comparisons
4-Person Analog Bridge vs 8-Person Analog Bridge	8
8-Person Speaker/Interrupter VCS vs 8-Person Analog Bridge	2
12-Person Speaker/Interrupter VCS vs 12-Person Analog Bridge	3
8-Person Speaker/Interrupter VCS with Delay vs 8-Person Analog Bridge with Delay	1
8-Person Speaker/Interrupter VCS with Delay vs 8-Person Simple Broadcast VCS with Delay	1
8-Person Simple Broadcast VCS with Delay vs 8-Person Analog Bridge with Delay	1
8-Person CVSD Bridge vs 8-Person CVSD Simple Broadcast with VCS	1
8-Person PTT Simple Broadcast vs 8-Person VCS Simple Broadcast – Interruptible	1

c. Experimental Procedure

Each experimental session is divided into two one-half hour periods. At the beginning of each period, the experimenter provides a short briefing which includes a description of the teleconferencing system to be studied during that period and the locations of telephones and telephone numbers to be used by conferees. He then answers questions, distributes materials required for solution of the conference scenario, and selects one person to serve as a "starter" for the session. Following this, conferees are released to locate their telephones and to initiate the dial-up procedure.

When the starter has verified that all persons have entered the conference and are able to communicate successfully with each other, the experimenter gives him or her a signal to begin. The starter informs the rest of the conferees that the signal has been given and the session is initiated. At this point, the experimenter begins monitoring the proceedings of the conference with the aid of headphones.

When, in the judgment of the experimenter, conferees have reached consensus that the best solution to the experimental problem has been found, or a period of 18 min. has elapsed since the "start" signal, the starter is instructed to inform conferees that 2 min. remain before termination of the session. When this latter period of time has elapsed, conferees are advised that the conference is over and that they should return to the main conference room.

When all conferees have returned, a short debriefing session is held during which subjects are told whether or not they achieved the optimal solution to the problem and, if not, what the nature of the optimal solution was. In addition, comments are solicited on the voice quality of the communication lines, special difficulties associated with interrupting other speakers or being interrupted by them, and any other comments pertinent to the use of the teleconferencing system. When the debriefing session is complete, orientation for the next session commences or the subjects are dismissed, depending on which half-hour period has just been completed.

It should be noted that the formality of the debriefing period has been slowly increased over the course of experimentation in order to take maximum advantage of the growing sophistication and insight of the subjects concerning subtleties of the various teleconferencing environments. Thus, where the primary content of debriefing sessions associated with early analog bridge experimentation was composed entirely of highly qualitative verbal description, almost all attitude and opinion data collected during research with the simple broadcast and speaker/interrupter VCS systems has been captured on written questionnaires and checklists in a form suitable for quantitative analysis. Examples of these instruments are included in Appendix A.

4. Preliminary Results

The success of efforts to conduct research on as many systems as possible has come at considerable cost to the statistical reliability that can be associated with the outcomes of various experimental comparisons. A glance at Table II-1 should indicate very clearly that only in the case of 4- and 8-person groups using the analog bridge does the number of replications approach what would be necessary for a high level of confidence to be associated with results to date. Nonetheless, we find a number of the observed outcomes to be particularly compelling and believe that they will prove to be durable in the face of repeated experimentation. Summaries of these outcomes are presented below.

a. Effects of Conference Size on Problem Solving

For teleconferencing systems and tasks of the type studied here, increasing the conference size from 4 to 8 or 12 persons produces little effect on the quality of conference performance, though it clearly has an impact on the speed and ease with which the conference can transact its business.

Support for these observations comes both from the comments of subjects and from comparisons of time and performance scores associated with information dissemination/collection and problem-solving phases of car pool. In the aggregate, these sources of information suggest the following:

(1) Differences in the times taken by the conference to complete each phase of the problem, differences in the times at which each member of a set of possible solutions is proposed, and differences in the orders in which solutions that vary in merit are proposed are accounted for almost entirely by the fact that the number of commuters to be dealt with in a given problem is perfectly correlated with conference size.

(2) Subjects report the adoption of more self-discipline in an effort to offset the increased likelihood of "collisions" and simultaneous speech in the larger conferences. To the extent that this self-discipline is evident in tape recordings of the sessions and in the computer-generated audit trails recently introduced into the data collection effort, we judge it to consist mainly in a requirement by the potential interrupter for longer pauses on the part of speakers before actual interruption, and for increased willingness to relinquish one's position as speaker if an interrupter is heard. Although this strategy is successful to a degree, it imposes a constraint on the free flow of communication that is perceived by conference members as a decrease in the ease with which the larger conferences can be conducted.

b. Effects of Teleconferencing System on Problem Solving

The type of teleconferencing system employed is clearly perceived by conference members to have an impact on the ease with which a conference is conducted. Quality and speed of solution, however, appear to be relatively independent of system type.

Results obtained with respect to the system variable are very similar to those obtained with the conference size variable, and may have a similar explanation, viz, that with slight shifts in communication strategy and with the adoption of some interactive discipline, experienced conferees overcome constraints imposed by the system. Where these efforts are successful, performance continues at a high level at some cost in ease of conferee interaction.

The best estimates made to date of the relative levels of difficulty experienced by conferees with the VCS and analog bridge systems, as well as with conferees of various sizes, have been obtained with the first two questionnaires included in Appendix A. These estimates are presented in Figs. II-8(a) and (b).

In Fig. II-8(a), the means of responses given by 12 experienced subjects who served in conferences containing 4, 8, and 12 conferees and who used both the voice control and analog bridge systems are plotted in normalized form. The distributions of responses whose means have been

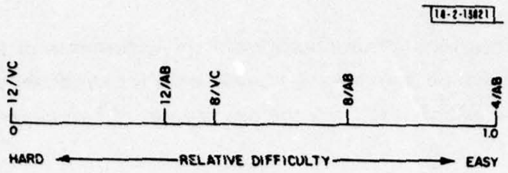


Fig. II-8(a). Normalized responses of 12 conferees plotted as distances along line of unit length. Differences between 12/AB and 8/VC are not significant.

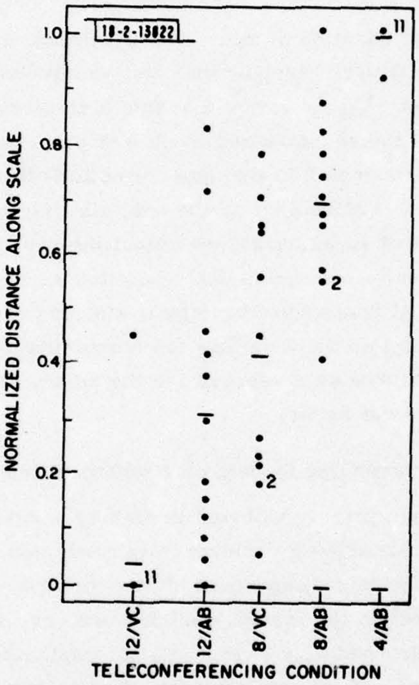


Fig. II-8(b). Dispersion of normalized responses. Horizontal lines represent means plotted in Fig. II-8(a). Numbers next to points indicate frequency of responses if more than one occurred for value given.

plotted are shown in Fig. II-8(b). Considering the total set of data represented here to consist of twelve replications of the ranking of five different conditions, a Friedman chi-square test estimates the probability that the observed distribution arose by chance to be less than 0.001.

c. Effects of Delay

The addition of a 500-msec delay between the generation of an input by a speaker and its receipt by other conference members initially creates an impediment to the free flow of conversation. The number of "collisions" rises significantly and speakers have a tendency to repeat themselves in an effort to assure themselves that they have been heard. The effect, however, appears to be transitory. Conferees quickly adapt to the delay and compensate for it by resorting to disciplines of the sort described earlier. The adaptation period is sufficiently short that the quality of the conferences shows no marked deterioration over that observed with undelayed systems, and conference "tempo" is only marginally slower than in those instances.

In Fig. II-9(a), the means of responses given by 7 experienced subjects who served in 8-person delayed and undelayed and VCS and analog bridge conferences are plotted in normalized form.* Corresponding distributions of normalized responses are shown in Fig. II-9(b). A Friedman chi-square conducted on the data of Fig. II-9 provides an estimate that the probability that the distribution of rank orders represented here arose by chance is less than 0.01.

On the basis of comments of the conferees, it appears likely at this time that a major reason for the increased difficulty experienced with transmission delays in both VCS and analog bridge systems has to do with the perceived interruptibility of a current speaker by a listener with something to say. Although responses to a question dealing with this factor have not yet been scaled, a test of rank orders replicating the distribution shown in Fig. II-9(a), but obtained with respect to ease of interruptibility, was significant at the 0.01 level.

5. Continuing Human Factors Efforts in Testbed Development

In the coming year, we expect to continue our efforts to evaluate a variety of teleconferencing systems and conference arrangements. The success of these further efforts will depend significantly on the satisfaction of certain methodological and experimental needs that have been identified during the past year. Among the most important of these are: (a) the need to develop an adequate statistical summary of computer-generated audit trails of the fine structure of the conference; (b) the need to compare results obtained with abstract laboratory scenarios against those obtained with realistic military scenarios. The remainder of this section will be devoted to brief summaries of our current thoughts on those items.

a. Statistical Summaries

As the result of a joint effort during the year by Lincoln and BBN, a statistical routine for the collection and aggregation of conference dynamics now exists as a permanent part of the testbed facility. This routine determines the status of each telephone line every 20 msec during the conference, sorts the outcomes into one of seven states, and, at the end of the experimental session, outputs a printed record of the accumulated states in both raw and aggregated form. Examples of portions of two audit trails and associated statistics are shown in Appendix B. An

* Responses of conferee 8 have not been completely analyzed to date, but rank order assigned to conditions are consistent with those of the other 7.

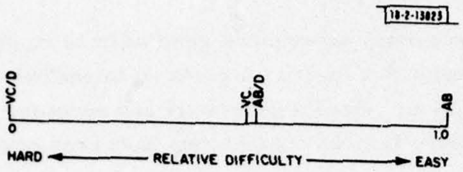


Fig. II-9(a). Normalized responses of 7 conferees plotted as distances along line of unit length. Differences between VC and AB/D are not significant.

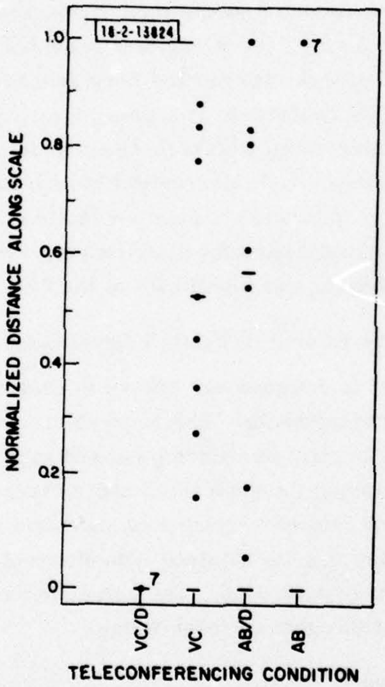


Fig. II-9(b). Dispersion of normalized responses. Horizontal lines represent means plotted in Fig. II-8. Numbers next to points indicate frequency of responses if more than one occurred for value given.

additional feature of this routine is that it permits the experimenter who is monitoring an ongoing conference to "mark" the trail by pressing a switch connected to one of the input channels whenever, for later analysis, he wishes to make note of a particular conference "event."

As it presently exists, the audit trail provides an in-depth look at conference dynamics and will, we expect, provide objective support for the attitude and opinion data collected from conference members. Moreover, it should prove to be invaluable as a diagnostic tool for the optimization of given teleconferencing systems and conference procedures.

Important work remains to be done on the statistics package before it can be used successfully in the evaluation program. A chief problem at the present time involves developing a satisfactory definition of "talk spurts," "pauses," and "floor time" in a context where optimum SNR thresholds may vary in different systems. As others have shown,[3] the "apparent" period of these and other parameters of speech will vary as a function of the threshold selected for the speech detection. The question for us, then, is how to compare systems that may have different thresholds with communication parameters that seem, of necessity, to be fixed.

b. Abstract vs Realistic Scenarios

To date, we have made the assumption that scenarios that would provide the most stringent test of systems would be those that would require a high degree of intercommunication and competition for the channel among members. The military conference, as we understand it to date, may not typically exhibit this characteristic. Indeed, pains are often taken to avoid the possibility of such competition through the use of priority schemes and other conference control procedures. As a consequence, we have resorted in this research to the use of fairly abstract problem-solving tasks.

It is clear that within the next phase of development and evaluation, we must test the correctness of the above assumption, and that our interest in conference scenarios must, to an extent, shift to environments modeled on those in which the systems of interest would actually be employed. In an effort to accomplish these goals, we have begun the specification of two different chairman roles to be studied within the context of large conferences (10 to 20 members), and the modification of the resource allocation scenario to allow for the incorporation of a priority structure and the use of conferencing protocols.

E. CONCLUSIONS AND RECOMMENDATIONS

The conclusions and recommendations presented here are based on our analysis of conferencing techniques, the results of the formal human factors experiments described in Sec. D above, and informal conference tests involving configurations not yet subjected to formal tests. Some of these informal tests were conducted using the facilities (described in the Semiannual Technical Summary[1]) which were limited to 4 participants. Because of the limited availability of formal experimental data, our conclusions should be viewed as preliminary. Formal experimentation is continuing to provide data in areas where our current conclusions are based on our subjective evaluation of informal tests.

1. Signal-Summation Techniques

a. The Analog Bridge

Conventional secure voice conferencing involves decoding the incoming speech signal at the conference controller, performing an analog summation of the signals, re-encoding the summed output, and transmitting that signal to all participants. If significant delays are present in the transmission and/or encoding process, it is necessary to prevent a speaker from hearing his own voice. This may be accomplished by subtracting out the speaker's voice from the sum signal sent to him. This technique results in a true full-duplex analog bridge such as that used in our experimental work. An alternative and widely used technique is to use echo suppressers in the lines between the participants and the bridge. This technique results in a half-duplex bridge action because the signal from the bridge to the participant is cut off whenever he speaks.

From a human factors point of view, the advantages of the full-duplex analog bridge for conferencing are:

(1) A participant will always be heard by the other conferees. The content of his utterance may not be understood if others are talking at the same time, but if PCM or other wideband waveform encoding is in use it is likely that it will be.

(2) Small comments, laughter, and the like can be heard by all without interference with the flow of the conference discussion. This property allows the participants to interact freely.

(3) The current speaker can be readily interrupted by some other participant who desires to do so.

(4) A participant can hear the others while he is talking and use their reactions to modify the presentation of his point, etc. if he so desires.

These advantages are confirmed by our formal human factors experiments which always show the full-duplex analog bridge to be superior to any signal-selection technique tested.

The disadvantage of the analog bridge for secure voice conferencing applications arises from the need to use narrow- and medium-band speech encoders in the transmission system. In the medium-band case, the extra decoding and encoding involved at the bridge reduce the perceived speech quality and make it more difficult to understand what is being said when two or more people speak at once. In the narrow-band case, the quality degradation caused by the tandemed encoding becomes severe and completely unusable when there is more than one active speaker. This difficulty is of theoretical origin because narrow-band encoders use a signal model based on the properties of a single speaker. They are inherently incapable of handling a mixture of voices.

b. Narrow-Band Signal Combining

If some technique could be found for combining narrow-band encoded-speech signals without converting to an analog representation, the tandemed encoding could be avoided and much of the advantage of the analog bridge could be retained for narrow-band speech. Unfortunately, there are no known techniques for effecting such a combination. We carried out an experiment involving rapid switching between two LPC-10 digital bit streams on a frame-by-frame basis in the hope that the human ear might be able to put together some sort of composite mixture of the two signals which would retain some of the speech content and/or speaker recognition properties of the individual encoded signals. This technique did not result in an effective fusion of the signals. In the absence of further ideas, we have concluded that narrow-band signal combining is not likely to be a viable technique for conferencing in the near future.

c. Majority Voting Bridge

For delta modulation encoding techniques, it is possible to approximate the effect of signal summation without converting to an analog representation. The bridging technique involves estimating the slope of the input signals by considering 2-bit sequences on the input data streams and producing an output bit stream which will result in a waveform having the slope characteristics of a majority of the inputs. The algorithm used in our experiments is described in more detail in Sec. C-1-b. Since the apparent SNR degrades as the number of inputs increases, we feel that use of the technique would be limited to conferences of at most 3 or 4 participants. To overcome this size limitation we developed a simulation which combines signal selection with the majority voting bridge in such a way that only the inputs exhibiting speech activity are combined in the bridge. This hybrid approach allows the technique to be used in any size conference with signal-to-noise characteristics which degrade only when more than one speaker is simultaneously active.

In one experimental session, we compared the majority voting bridge with a simple broadcast switched conference using the same 16-kbps CVSD encoding. The subjects indicated a slight preference for the switched system over the bridge, but the experiment requires replication before any significance can be attached to the result. Our own subjective listening suggests that the majority voting bridge loses much of the advantage of the analog bridge because the quality deteriorates badly when two or more speakers are active. In that case, it is much more difficult to understand what is being said. In addition, occasional noises from other inputs will cause degradation of the signal in cases where only one person is speaking. It may well be possible to improve the performance of the majority voting bridge by further refinement of the speech-activity detector action, but we suspect that even in that case it would not have any significant advantage over a signal-selection technique.

d. Conclusion – Summation Techniques

Even though the analog bridge is the best conferencing technique for wide-band encoding, the absence of any bridging technique for narrow-band encoding forces us to choose some form of signal selection for military systems which must accommodate narrow-band users.

2. Signal-Selection Techniques

a. Control Inputs

There are three basic techniques for controlling a signal-selection conferencing system. In the terminology of Sec. C-2, these are:

(1) Voice Control Switching (VCS),
(2) Push-to-Talk (PTT), and
(3) Control Signal Switching (CSS).

VCS is the most natural of these techniques, requiring no explicit action on the part of a would-be speaker other than starting to talk. Experience has shown that people can readily learn to modify their speaking behavior to make effective use of VCS systems. The only major difficulty with VCS techniques is sensitivity to noise. When background noise level is high, the speech-activity detectors may switch speakers at inappropriate times or fail to switch at all if a steady noise of sufficient amplitude is present at some input. In real-world applications, a

VCS system must have some sort of protection against such noise problems. Probably some combination of adaptive activity thresholds and local PTT buttons in extremely noisy environments would suffice. We have not addressed these problems in our experimental program where the participants are in relatively quiet office environments.

PTT is somewhat less natural to use than VCS, but can be learned without great difficulty, as witnessed by the many users of half-duplex radio communications systems. An inexperienced user will occasionally forget to push the PTT switch when starting to speak, or will release it too soon and clip off the end of the utterance. PTT works well in noisy environments and can be used to implement conference controllers which switch encrypted bit streams, thereby avoiding the necessity of securing the controller. (VCS can have a similar security advantage if speech-activity detection is done at the input encoder and the resulting signal is communicated in the clear to the conference controller.) PTT need not be used in the half-duplex form in which it is typically observed, and was used in our experiments to date. We expect that a full-duplex PTT conferencing system could have performance characteristics very similar to those of a VCS system following the same basic switching algorithm. In general, a PTT system will be less interruptible than a VCS system, since it is easier for a participant to keep holding the switch closed than to keep talking without pausing.

CSS requires yet more learning on the part of the participants, and is likely to be unsuited to occasional users of a conferencing capability. However, it can provide smoother interaction in situations involving large delays and distributed control of a shared communication channel by avoiding channel contention with consequent loss of speech to all participants. In addition, the often contradictory demands of priority and urgency can be dealt with by using different types of request-to-speak signals. Our experience with a 4-participant, 2-button, and 2-indicator-light CSS system has demonstrated a basic difficulty with the CSS technique. In simple sequential conference situations in which one person raises a question for which another has the answer, the system may switch to some third person who had wanted to say something on a different topic and had pushed his want-to-talk button in anticipation. If he now raises his new topic, he will tend to disturb the flow of the discourse. If, instead, he aborts his request, he will not divert the discourse but will have slowed the arrival of the desired response to the original question. In both cases, the effectiveness of the conference will have been reduced somewhat.

b. Switching Algorithms

The switching algorithms considered in this study are those suited to a central conference controller and limited to a single full-duplex (4-wire) communications link between each participant and the controller. With these limitations, the following design decisions remain to be made to specify a conferencing system:

(1) What will the selected speaker hear while he holds the floor? There are two reasonable choices: (a) nothing (simple broadcast), or (b) some other participant who happens to talk (speaker/interrupter).

(2) Under what conditions can a selected speaker be interrupted? There are a range of possibilities here. We have considered two: (a) the speaker can continue until he finishes (the noninterruptible case), and (b) the speaker can be interrupted easily at any time.

Obviously, priority and urgency are factors which can affect the question of interruptibility. We have not yet explored these factors in our experimental work. Priority can be readily included in a VCS conferencing system, but to explore its effectiveness requires different scenarios than those used so far in our human factors experiments. To signal the urgency of a request to talk requires a more general signaling capability than can be realized with VCS. We feel that the CSS technique is needed to convey urgency if priority is also in effect.

(1) What the Speaker Hears

In our judgment, it is important for the communication link between the conference controller and the selected speaker to be active, even though most of the time the speaker hears nothing while he is talking. In a simple broadcast system, the presence of speech in his receiver while he is talking is a sure indication that a talker has not been selected as the conference speaker. The absence of speech in his receiver is a fairly good indication that he has been selected as the speaker, but there is always the possibility that his communications with the conference have failed. In this respect, the analog bridge technique is superior if truly full-duplex (no echo suppressors). Also in this respect, the speaker/interrupter technique is less satisfactory since a talker may be the selected speaker even though he hears another voice (the interrupter). However, he cannot be sure whether the voice he hears is that of the interrupter or the speaker, in which case he may be either the interrupter or a third talker who is not being heard by anyone. Of course, if he hears nothing he can generally assume that he is speaking to the conference. We found that many of our experimental subjects, once made aware of this feature of the system, used it to discover whether or not they were being heard by the others.

(2) Interruptibility

We have conducted a limited set of experiments to explore the interruptibility issue. The systems compared were (a) simple broadcast – noninterruptible (both VCS and PTT), (b) simple broadcast – interruptible (VCS), and (c) speaker/interrupter (VCS). Section C-2 describes these switching algorithms in detail. The subjects did not express strong preferences among the three VCS systems. The PTT was less favorably received because of its half-duplex feature, which is not related to the interruptibility question. The subjects exhibited a slight preference for the noninterruptible simple broadcast over the interruptible, and were undecided between the interruptible simple broadcast and the speaker/interrupter system. We think that they tend to prefer the noninterruptible systems because they rarely have a real need to interrupt a speaker in the problem-solving scenarios which they have been using. The scenarios do not lead to long monologues which would motivate a desire to interrupt. It is easier and just as effective to wait for the speaker to finish.

It is interesting to listen to the "interrupter" channel of the speaker/interrupter algorithm during an experimental session. One does not often hear utterances meant as interruptions and directed to the current speaker. Rather, one hears utterances and fragments of utterances meant for the conference as a whole. These get switched to the interrupter channel because the speaker channel is occupied. Such "interruptions" are largely the result of collisions which occur when two or more speakers attempt to start talking at the same time. Such collisions occur a number of times during a typical experimental conference because the scenario produces situations where two or more participants have alternative answers to questions which are raised. In the noninterruptible system, the listening participants are likely to be able to hear one of the responses to the question without much difficulty. In the interruptible case, it is more likely that the collision will end up destroying the intelligibility of all responses, requiring more repetitions to get the information transmitted successfully.

It should be pointed out that our noninterruptible VCS technique is not truly noninterruptible as is the PTT technique. The interrupter must wait for the speaker to pause for a time of the order of one-half second before an attempted interruption can be successful. In our judgment, the VCS "noninterruptible" broadcast technique has sufficient interruptibility for use in most conferencing situations. We suspect that the PTT noninterruptible broadcast technique would not be satisfactory. We have not yet conducted experiments to test this hypothesis.

For informal conferences such as those of our experiments, the speaker/interrupter technique does not appear to offer any substantial benefit, and further testing may well show that it has measurable disadvantages. For formal conferences, however, we suspect that making the interrupter path available to the chairman could facilitate his control of the conference, particularly if a PTT technique were in use. We plan to conduct experiments to test this hypothesis.

c. Conclusions – Switching Techniques

Although the results of the human factors tests reported in Sec. D-4 above indicate that subjects always prefer the analog bridge technique to any signal-selection technique, they also show that subjects can adapt readily to signal selection in such a way as to accomplish the conference task in about the same overall elapsed time. This latter result is important because it shows that a signal-selection technique can be used without impairing the participants' performance in conferencing.

The simple noninterruptible broadcast technique is adequate for informal conference situations. The VCS speaker/interrupter technique may be superior for formal chaired conferences.

PTT techniques can be used to advantage in noisy environments. PTT techniques should operate in a full-duplex mode so that a talker can determine whether or not he is being heard.

3. Delay Effects

As might be expected, the introduction of communication delays of the order of one-half second (two satellite hops) makes conferencing more difficult for the participants. Some time was required for the participants in the formal experiments to adjust their style to the presence of delay when they first experienced it. Delay makes all conferencing techniques more difficult to use. The rapid interaction observed with the analog bridge disappears when delay is introduced, and subjects behave in a more formal fashion similar to that which they adopt for a signal-selection technique. With signal selection, delay prolongs the duration of collisions because a would-be speaker does not get immediate feedback as to whether or not he is the selected speaker. Conferees quickly adapt to the delay and compensate for it.

We have not observed any interaction between delay effects and conferencing techniques. That is, we have not seen any instances where technique A was preferred to technique B without delay, and the reverse became true when delay was introduced.

4. Conference Size Effects

Our formal experiments confirm the expected effect that conferencing becomes more difficult as the number of participants increases. With the exception of the majority voting bridge in its pure form, none of the techniques we have explored have any inherent size limitations.

Signal-selection techniques have a theoretical advantage over summation techniques in that, with selection, there is no increase in background noise as size increases. Within the range of our experiments to date (12 participants), bridge noise is not a problem.

As in the case of delay effects, no interaction has been observed between conference size and technique.

5. Speech-Encoding Techniques

While most of our formal experiments have used high-quality PCM speech, earlier work with a 4-participant, push-button-controlled conferencing system involved LPC and CVSD speech encoding. We conclude from that experience that, for signal-selection techniques, there is not likely to be any interaction between encoding techniques and conferencing algorithms. Of course, conferencing will be increased in difficulty as voice quality decreases with bandwidth and/or tandeming.

6. Recommended "Best" Technique

For a conferencing system which has only analog or wideband PCM users, the conventional analog bridge is recommended. For a system which must accommodate narrow-band users, a voice-control switched conference configuration is recommended. For chaired conferences, the interrupter channel can be totally pre-empted by the chairman to function as an order wire for the chairman to signal to the broadcaster. For unchaired, free-for-all conferences, the interrupter channel may or may not be useful.

We now consider system requirements for the recommended speaker-interrupter system (S-I) and a baseline tandem analog bridge system as well as a PTT simple broadcast conference.

For remote terminal equipment, a full-duplex speech encoder-decoder with appropriate cryptographic capability is assumed adequate for the S-I and tandem analog bridge systems. For the PTT system, an additional control signal is needed for the talk control. If we assume a central bridge control, the talk control might be integrated with the voice transmission. At any rate, terminal requirements for our S-I system are met by deployed full-duplex DCS equipment.

For control equipment at the central bridge, the three approaches have distinct requirements. The tandem analog bridge requires full receiver equipment for each conferee, a red analog summing node, and an additional transmitter encoder-encrypter for each conferee. We assume that network timing signals would be available for any system. Basically then, a tandem red analog bridge central conferencing node would require N sets of terminal equipment for N conferees, and each line from bridge to conferee must deal with the echo-suppression problem. For the simplest broadcast PTT conference assuming the talk control is encoded in some form on the voice stream (any other approach seems to require even more complex signals), the central conference bridge must process each conferee digital stream to get at the encoded voice stream. Special-purpose bridging equipment would scan each digital voice stream (e.g., 8-kbps APC) for the talk control signal. This scanning function requires frame synch computation. With appropriate priority or chairman pre-emption rules a detected talk request would be honored by transmitting the selected digital speech signal out through N (or N – 1) encryption equipments back to the conference. This bridge equipment would be considerably less complex than the full tandeming analog bridge since each of N digital voice streams does not require a voice decoder to produce analog output to a bridge. In addition, the bridging functions required can be realized in a bridge device working at the slower frame rates (e.g., 20 msec).

Finally, our recommended S-I system is no more complex in central bridge control equipment than the simple PTT system. At the central control, each conferee stream must be received to the digital voice stream level and energy detected on each stream. No special talk signal must be included in the stream from each conferee. The bridge control honors a conferee above threshold after applying priority and chairman weighting. The broadcaster digital voice stream is then output to the N – 1 transmitter crypto units. A second interrupter talker can be detected by the bridge at the digital voice stream with slight additional logic complexity and output to the remaining unused crypto-transmitter unit. It appears that the recommended S-I system is less complex than the simple PTT conference, since the S-I system needs no special PTT transmitted control. In addition, both of these approaches are far less complex than the tandem analog bridge, as well as producing improved untandemed voice quality.

From the point of view of priority structuring and chairman roles, the S-I system control bridge and interrupter channel allows for considerable flexibility.

III. SPEECH ALGORITHM STUDIES

A. INTRODUCTION

The effort in speech algorithm studies during most of FY 77 focused on the problem of distortions caused by wide dynamic range talkers using LPC vocoders. We felt that an enhanced useful dynamic range for LPC vocoders would improve not only the quality of the LPC vocoder used alone, but the more critical tandemed configurations of LPC-CVSD and CVSD-LPC that occur when internetting, or remote conferencing.

A second area of interest in this effort was in gaining improved LPC performance via the inclusion of some information relating to the residual error signal. Since the LPC spectrum analysis process is best matched to a "poles-only" speech model, sounds for which that model may be inadequate are reproduced with poor fidelity. The LPC residual is composed of the difference between the actual speech signal and one whose spectral envelope is that which has been determined by the LPC process. The residual therefore contains any information relating to vocal tract zeros that may not have been adequately handled in the LPC algorithm. Direct transmission of the LPC residual is overly consumptive of bandwidth however, and its inclusion in the output of a narrow-band speech terminal would violate overall bit-rate constraints. Section C below describes an attempt to apply channel vocoding techniques to the LPC residual waveform in the hope of compressing its transmission bandwidth requirements to an acceptably low level. Our effort was primarily dominated by implemental difficulties due to equipment limitations, and, as a result, no conclusive evaluation of the basic concept has yet emerged.

B. LPC AGC ALGORITHM

At first glance, it might seem that the obvious solution to the problem of distortion caused by wide dynamic range input would be to precede the A/D converter with a conventional fast-attack, slow-decay AGC circuit. This approach is undesirable for at least two reasons. First, such an AGC circuit tends to produce speech of relatively uniform volume, which is both unnatural sounding and tends to boost the background noise level when a soft speaker is talking. Second, this form of AGC inevitably distorts the speech waveform presented to the vocoder for analysis, thus producing unpleasant artifacts in the resultant synthetic speech. It is for these reasons that attention has been directed at developing a new AGC strategy that will circumvent these difficulties while at the same time increasing the input dynamic range of the LPC algorithm.

The distortion caused by loud speakers can be cured by adjusting the input gain to the A/D converter so that the loudest speaker the LPC algorithm will be expected to tolerate does not cause A/D converter clipping. Normal and soft speakers will now use only a fraction of the available A/D converter dynamic range and thus cause a possible degradation of the LPC quality. This loss of quality can be due to two causes: increased input quantization noise, and loss of significance in the LPC analysis calculations. The quantization noise problem can only be cured by using an A/D converter with a larger word size, or using a program-controlled attenuator at the input to the usual 12-bit A/D converter. One important result of the present investigation is that increased input quantization noise does <u>not</u> seem to cause degradation of the LPC speech quality, thus obviating the need for these hardware cures.

The problem of loss of significance in the analysis calculations can be attacked by suitably upscaling the input speech as it comes from the A/D converter. The proper way to do this is to uniformly scale up each frame of speech as much as possible without causing overflow. Since

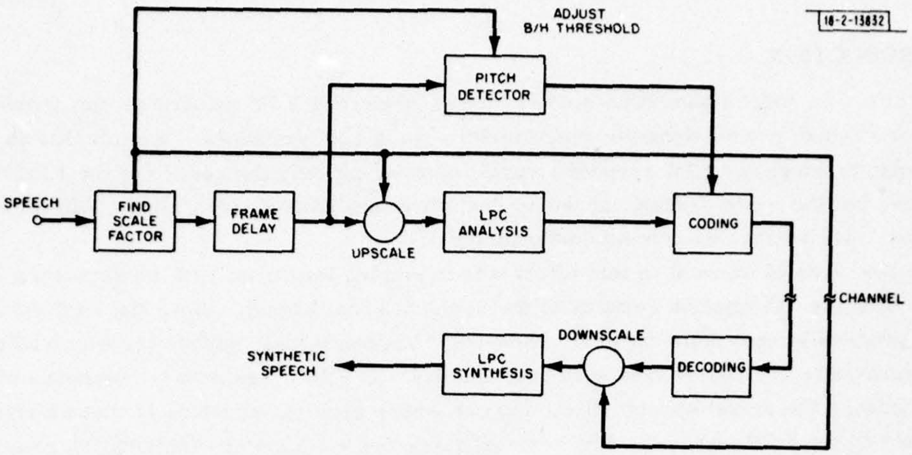


Fig. III-1. Experimental AGC system.

the LPC analysis is inherently frame oriented, this scaling strategy does not distort the speech waveform even though amplitude discontinuities are produced at frame boundaries. A block diagram of the scheme used to perform this scaling is shown in Fig. III-1. The scale factor for a given frame is determined by first finding the maximum rectified speech sample in the frame. The number of bits (up to a maximum of 3 bits) that this maximum sample can be left-shifted without overflow is the scale factor for that frame. The incoming speech is delayed by one frame so that the scale factor just determined can be applied to each sample in that frame. (It will be shown later that this delay can be eliminated.) The scaled speech is now analyzed with the conventional LPC algorithm, after which the parameters are coded and shipped to the receiver. An additional 2 bits describing the scale factor used for the frame are sent along with the other parameters.

There is one further problem that must be addressed; namely, the behavior of the Gold-Rabiner pitch detector when subject to low-amplitude input signals. This pitch detector uses an energy measure to make a preliminary decision as to whether a frame is voiced or unvoiced. An input speech frame of sufficiently low energy, as measured by comparison with an empirically determined fixed threshold, is automatically declared unvoiced. This means that the low-amplitude signals that the AGC algorithm is attempting to recover will be declared unvoiced. In order to combat this effect, an attempt was made to use the upscaled speech as the input to the pitch detector. This led to annoying artifacts in the synthetic speech most likely due to the fact that the Gold-Rabiner pitch detector, unlike the LPC analyzer proper, is not frame oriented and so responds adversely to amplitude discontinuities at the frame boundaries.

The next attempt at combating the problem was simply to reduce the energy threshold while at the same time using the unscaled speech as the input to the pitch detector. The threshold was reduced a fixed number (3 bits) corresponding to the maximum allowed scaling of the speech going to the LPC analysis. This strategy, in conjunction with the rest of the AGC algorithm, seems to work very well as judged by careful listening tests.

It should be noted that the excitation amplitude is the only parameter whose value has been affected by the upcoding operation. The pitch and reflection coefficients are independent of the amplitude of the input speech. The excitation amplitude must be downscaled, otherwise the resulting synthetic speech would be too loud and unnatural for comfortable listening.

After reception, the parameters are decoded and the excitation amplitude for the synthesizer filter is downscaled by an amount based on the frame-scale factor. The remainder of the receiver processing is the normal LPC synthesis algorithm.

The downscaling operation has been the subject of intensive investigation. There are two issues here: where in the analysis-synthesis chain to perform the downscaling, and what strategy to use for downscaling. Three places for downscaling suggest themselves – before coding, after decoding as shown in Fig. III-1, and after synthesis. All three ideas were attempted, and downscaling after decoding was judged to be the best method. Downscaling before coding worked well with high-level input signals, but performed poorly with low-level inputs because the residual energy was too low in the coding table. Downscaling after synthesis was rejected because it yielded a "bumpy" sounding output due to frame-to-frame discontinuities in the synthetic speech.

The first downscaling strategy that was tried was a linear one; i.e., the excitation amplitude was downscaled by the same number of bits that the input speech was upscaled by in the current frame. This algorithm produces speech that is indistinguishable from ordinary LPC speech until the input level drops to the point where ordinary LPC drops into steady hiss. Both algorithms are still perfectly intelligible at this point. This state prevails until the input level becomes too small for the coding table. The AGC version of the algorithm is still highly intelligible, but the output level is too low to be useful.

The downscaling strategy that was finally adopted is a nonlinear one; i.e., the excitation amplitude is downscaled by an amount dependent on, but not equal to, the amount that the input speech was upscaled. This scaling algorithm is depicted in Fig. III-2 where it can be seen that it compresses amplitudes at low and high levels, but is linear at intermediate levels. Listening tests with this algorithm have met with very favorable results. At input levels low enough to cause ordinary LPC to suffer severe energy quantization effects, the AGC version of the algorithm still produces excellent quality speech at an acceptable output level. Typically, participants in such tests comment on how "easy" or "relaxing" the vocoder is to use because one need not worry about the level of one's voice.

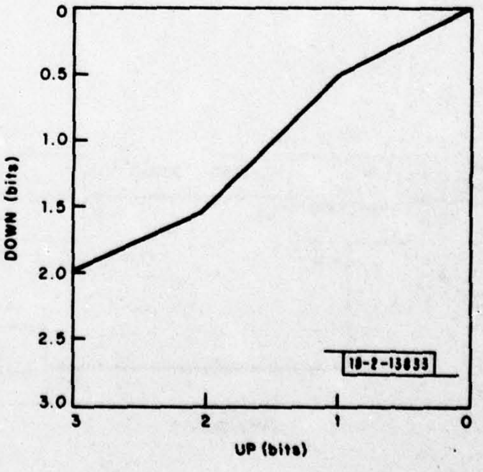


Fig. III-2. The nonlinear downscaling algorithm.

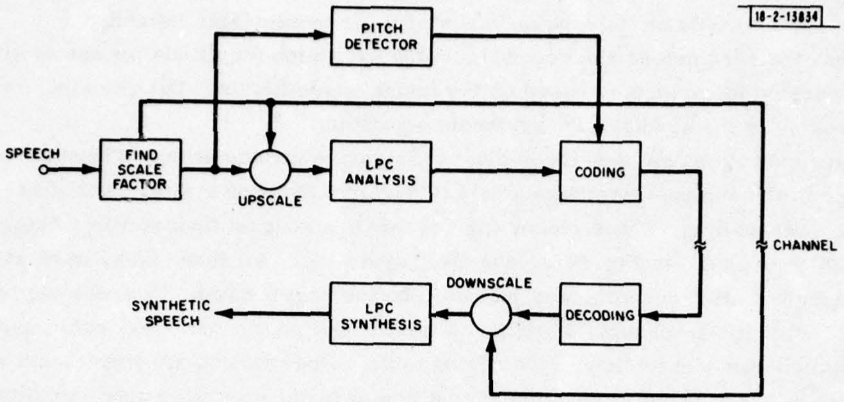


Fig. III-3. Final LPC AGC system.

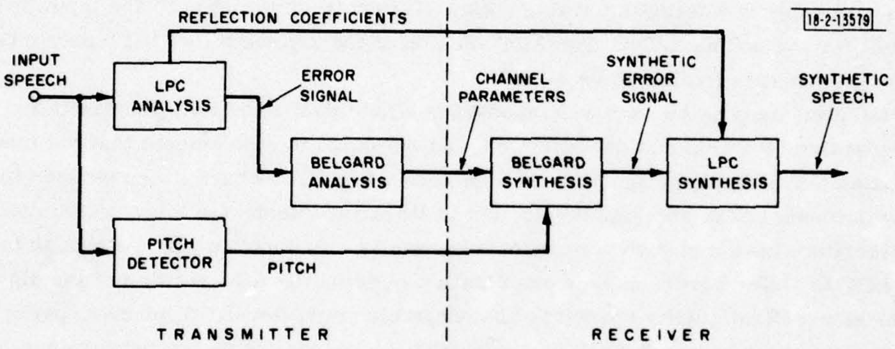


Fig. III-4. Proposed vocoder algorithm.

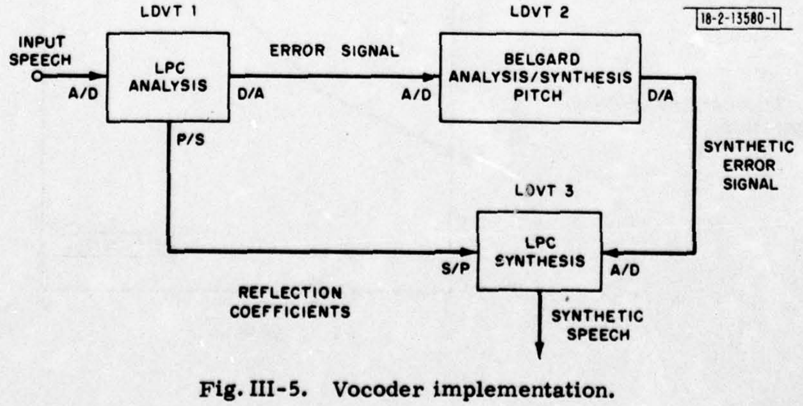


Fig. III-5. Vocoder implementation.

The only apparent drawback of the AGC algorithm just discussed is that it requires the buffering of one complete frame of raw speech. Accordingly, an experiment was run in which this buffer was removed. In this configuration, a given frame of speech is scaled using the scale factor for the previous frame. The measured correlation between the maximum speech samples in successive frames was sufficiently high (approximately 0.95) to indicate that this relaxed procedure might work. The scaling operation in this configuration can now produce overflow when applied to certain of the samples in the frame. In order to minimize the audible effects of such overflows, saturation arithmetic is used during scaling which has the effect of replacing overflow by clipping. Listening tests of this version of the algorithm have concluded that its performance is indistinguishable from the earlier algorithm employing a frame delay. A block diagram of the final algorithm is shown in Fig. III-3.

C. LPC-BELGARD EXPERIMENT

Most of the following material appeared in the last Semiannual Technical Summary,[1] and is included here for completeness. A new type of vocoder algorithm has been proposed to try to overcome the limitation that a conventional LPC vocoder cannot model spectral zeros. A block diagram of this algorithm appears in Fig. III-4. The basic idea is to use LPC techniques to generate a residual error signal which is then analyzed by a channel vocoder filter bank. Since a channel vocoder is capable of modeling both the poles and zeros of its input, this artifice yields a method of introducing zeros into the standard LPC modeling of the vocal tract. The parameters shipped to the receiver are the LPC reflection coefficients, the channel vocoder parameters characterizing the spectral envelope of the error signal, and pitch derived from a pitch detector working directly on the input speech. At the receiver, the pitch word is used to generate the excitation for the channel vocoder synthesizer whose output should be an approximation to the error signal. This synthetic error signal is now used to excite a conventional LPC synthesis filter whose output is then the final synthetic speech.

The implementation of this algorithm on a LDVT is a nontrivial task because the Belgard algorithm alone uses most of the LDVT's resources, both with respect to running time and memory occupancy. This means that it is impossible to run the algorithm in a single LDVT. Using two LDVTs, one for LPC analysis/synthesis and another for Belgard analysis/synthesis, is also not feasible because it requires the LDVT running the LPC algorithm to input two samples and output two samples every 132 μsec. The LDVT's limited I/O capability makes this impossible to do. These considerations led to the experimental setup shown in Fig. III-5. LDVT1 will accept the input speech and use LPC analysis to produce the residual error signal. The latter will be shipped to LDVT2 via LDVT1's D/A port where it will be subjected to Belgard analysis/synthesis. Belgard's pitch detector will be used to derive the pitch from the error signal rather than from the raw speech, as was discussed earlier. The resulting synthetic error signal will be shipped to LDVT3 via LDVT2's D/A port. In addition, LDVT1 will serialize the reflection coefficient parameters and ship them to LDVT3 via the parallel/serial-serial/parallel (P/S-S/P) interface. These parameters will then be unpacked and used in conjunction with the incoming synthetic error signal from LDVT2 to produce the final synthetic speech.

This arrangement was implemented, and it was determined that since use of the serial channel to ship the reflection coefficients to LDVT3 requires buffering, rate control, and synchronization protocols, time-varying relative delays between the reflection coefficients and the synthetic error signal are introduced. These delays produce unacceptable artifacts in the output speech. These artifacts are present even when the raw error signal produced by LDVT1 is shipped directly to LDVT3 without Belgard intervening.

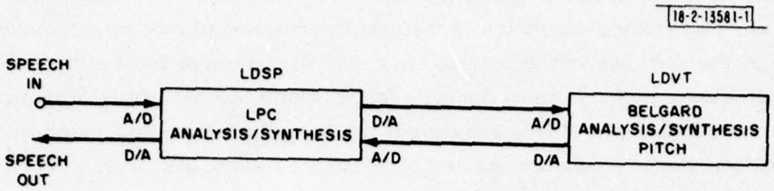


Fig. III-6. Proposed vocoder implementation.

The Lincoln Digital Signal Processor (LDSP) is an advanced version of the LDVT that offers a possible solution to the implemental difficulties outlined above. Construction of the LDSP was completed during FY 77, and its associated A/D and D/A conversion peripheral was installed late in the fiscal year. With several simple modifications, the LDSP system can be configured to accept two analog input streams and deliver two analog output streams. With this capability, the algorithm might be tested with the setup shown in Fig. III-6 which is algorithmically the same as that of Fig. III-5 except that now LPC analysis and synthesis are done in one machine (the LDSP), thus eliminating the need for the serial data path and its attendant delay problems.

IV. BANDWIDTH EFFICIENT COMMUNICATIONS

A. INTRODUCTION

Investigations into techniques for efficiently integrating voice and data traffic in digital communications systems have continued at Lincoln during FY 77.

A Slotted Envelope NETwork (SENET) voice/data multiplexer has been simulated on the PDP 11/45 facility, in order to investigate the effects on data performance of fluctuating numbers of voice users. The results on data-packet queue lengths and delays disagree with the predictions of a previous analysis.[4] A description of the simulation and an analysis of the results are reported here.

The Packetized Virtual Circuit (PVC) scheme for integrating voice and data, a model of a single link in a PVC network, and a computer simulation of this model are described in detail in the last Semiannual Technical Summary[1] and in other references.[5-7] A brief summary of this work is presented here. Also, an initial investigation of a hardware implementation for a PVC nodal switch is presented to provide some means to compare size, complexity, and cost with other known network nodal switches.

The simulation study of the PVC network has focused on the effects on data-packet delay of variations in voice traffic due to talk spurts and silences of individual talkers, for a fixed number of ongoing conversations. One of the problems not previously addressed in the PVC simulation is the effect of fluctuations in the number of voice users as calls are initiated and terminated. This problem has been considered previously in the context of the SENET technique for the integration of voice and data.

B. SIMULATION OF DATA PERFORMANCE IN A SENET MULTIPLEXER

A SENET concept for integration of circuit-switched (voice) traffic and packet-switched data traffic has been presented by Coviello and Vena.[2] Performance of the SENET multiplex structure has been analyzed by Fischer and Harris.[4] The SENET multiplexing scheme is based on a time-division multiplex (TDM) frame structure where a certain number of the time slots in the frame are allocated to circuit-switched voice traffic. Initiation of a voice call results in one of these time slots being assigned for the duration of the connection; if all S voice slots are busy at call initiation, the call is blocked and cleared from the system. At a given time, a number $n_v \leqslant S$ voice slots will be in use. Data traffic is assigned to the remaining N slots in the frame. However, a "movable boundary" is allowed where data traffic is permitted to use any of the $S - n_v$ voice slots which may be momentarily free. To maintain low blocking probability, S must be larger than the average $\overline{n_v}$ of the number of voice connections. Thus, on the average, one would expect $S - \overline{n_v}$ residual slots to be available for data. Silence detection is not exploited in the SENET system, so the fluctuations in voice traffic are due to initiation and termination of calls. The effects of this variation were not included in the PVC simulation, which assumed that silence detection was exploited and dealt in detail with the effects of individual talkers switching between talk spurt and silence.

A simulation of data performance in a SENET "movable boundary" data and voice multiplexer is described here. The conclusion indicated by the simulation results is that an attempt to utilize for data a substantial portion of the average residual capacity of the voice segment of the frame, will result in very large data-packet queues and delay. This conclusion disagrees with the results of a previous analysis.[4] The reasons for this discrepancy are d'scussed below.

The results of the simulation indicate that large buildups of data-packet buffers and delays occur during periods where the voice channel occupancy is high, if the allowed input data flow is fixed according to the long-term *average* of data capacity. This problem points up the need for a flow-control mechanism to lower (or raise) the allowed rate of data flow into the multiplexer with time variations of voice channel occupancy. The simulation model and results are now described, beginning with a discussion of voice traffic behavior and then proceeding to the results on data queue sizes and delays. The need for flow control to prevent large buffer buildups will then be discussed.

1. Voice Traffic Simulation Model

Voice customer arrivals are modeled as a Poisson process with average rate λ customers/sec. The call holding time distribution is taken to be exponential with mean μ^{-1} sec. The average number of call arrivals λb in a $b = 10$-msec frame interval is generally small enough so that the TDM framing structure can be ignored for voice traffic analysis. Thus, the voice multiplexer can be described as a classical S-server loss system M/M/S/S (see pp. 105-106 in Ref. 8). The behavior of this system is governed by the state-transition-rate diagram shown in Fig. IV-1. The ovals in this diagram indicate system states, where state k represents the event that n_v, the number of calls in progress, is equal to k. From this diagram, one can determine the steady-state probabilities p_k that $n_v = k$ at a random time as the "erlang-B probabilities"

18-2-13825

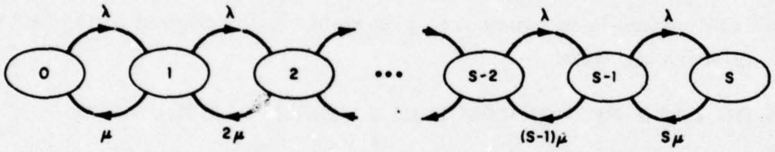


Fig. IV-1. State-transition-rate diagram for S-server loss system M/M/S/S. When system is in state k, k calls are active and k voice plots are occupied $[n_v(t) = k]$.

$$p_k = \frac{(\lambda/\mu)^k/k!}{\sum\limits_{k=0}^{S} (\lambda/\mu)^k/k!} \qquad k = 0, 1, \ldots, S \quad . \tag{IV-1}$$

Here, p_S is the probability that a random call will find all servers busy and be blocked. The average number of active calls is

$$\overline{n_v} = \sum_{k=0}^{S} kp_k = \left(\frac{\lambda}{\mu}\right) (1 - p_S) \quad . \tag{IV-1a}$$

A dynamic simulation of the variation of n_v with time, based on the state-transition diagram, has been developed as follows. Assume that $n_v = k$ at a particular starting time. Then, n_v is held at k for a time τ drawn from an exponential pdf

$$f_{\tau_k}(\tau) = \tau_k^{-1} e^{-\tau/\tau_k} \qquad \text{(IV-2)}$$

where the mean holding time τ_k is determined as

$$\tau_k = 1/(\lambda + k\mu) \qquad k = 0, 1, \ldots, S-1$$

$$\tau_S = 1/S\mu \quad . \qquad \text{(IV-3)}$$

After a time τ, n_v is increased to k + 1 with probability

$$p_{up}(k) = \lambda/(\lambda + k\mu) \qquad k = 0, 1, \ldots, S-1$$

$$p_{up}(S) = 0 \qquad \text{(IV-4)}$$

or decreased to k − 1 with probability $1 - p_{up}(k)$. This process is repeated as often as desired to generate sample functions of $n_v(t)$.

2. Data Traffic Simulation Model

Data packets are assumed to arrive in a Poisson process with rate Θ packets/sec. Packet size is assumed fixed and equal to the number of bits in one TDM slot. Following Ref. 4, this slot size has been taken to be 80 bits. During any frame, the number of slots available for transmission of data packets is $N + S - n_v$, the sum of N dedicated data slots and $S - n_v$ unused voice slots.

Consider a period of time of duration t_i between the i^{th} and $(i+1)^{st}$ transitions in $n_v(t)$, during which $n_v(t)$ = constant = $n_v^{(i)}$. The number n_+ of data packets arriving during this interval is drawn from a Poisson distribution

$$p(n_+) = \frac{(\Theta t_i)^{n_+} e^{-\Theta t_i}}{n_+!} \qquad \text{(IV-5)}$$

with mean Θt_i. The maximum number of data packets which can be sent out during this time is

$$n_- = [N + S - n_v^{(i)}]\,(t_i/b) \quad . \qquad \text{(IV-6)}$$

(Since generally $t_i \gg b$, the fact that t_i/b is not necessarily an integer has been ignored.) Let the number of data packets waiting for service at the beginning of this i^{th} interval be $n_d^{(i)}$. The evolution of n_d to the next interval is simulated as

$$n_d^{(i+1)} = \max[n_d^{(i)} + n_+ - n_-, 0]_2 \qquad \text{(IV-7)}$$

where the max insures that n_d never becomes negative. This equation is strictly valid as long as the data queue does not empty during the interval t_i. If the data queue does empty during that interval, then $n_d^{(i+1)}$ may be slightly greater than predicted by Eq. (IV-7) if the dispersion of packet arrival times causes some packet service slots to go unused. This slight discrepancy during periods when n_d is close to zero would have negligible effects on the results presented below, and has been ignored.

The model used for the combined voice/data simulation is described by Eqs. (IV-2) through (IV-7). Note that n_d is updated only at times of transition in $n_v(t)$. For the purpose of display and of determining time averages, $n_d(t)$ is assumed to vary linearly between its sample values $n_d^{(i)}$. Suitable random-number generators are used to determine τ_k, n_+, and the up/down transition decisions for $n_v(t)$. The average data packet waiting time is determined from measured values of $<n_v>$ by Little's theorem as

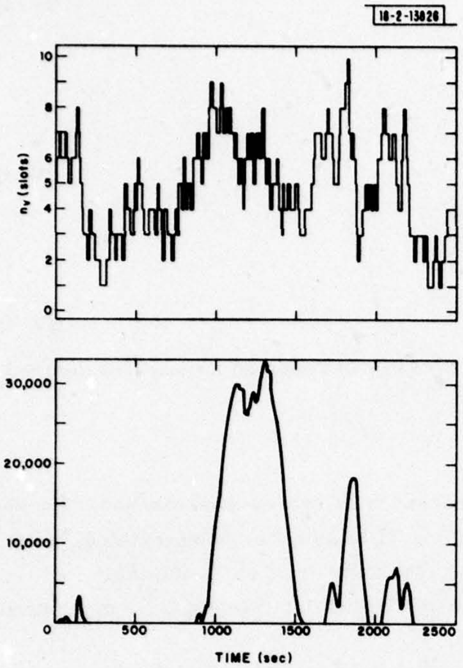


Fig. IV-2. Sample functions of $n_v(t)$ and $n_d(t)$ for $S = 10$, $N = 5$, $\lambda = 0.05\ \text{sec}^{-1}$, $\mu = 0.01\ \text{sec}^{-1}$, and $\Theta = 900$ packets/sec ($\Theta b = 9$ packets/frame).

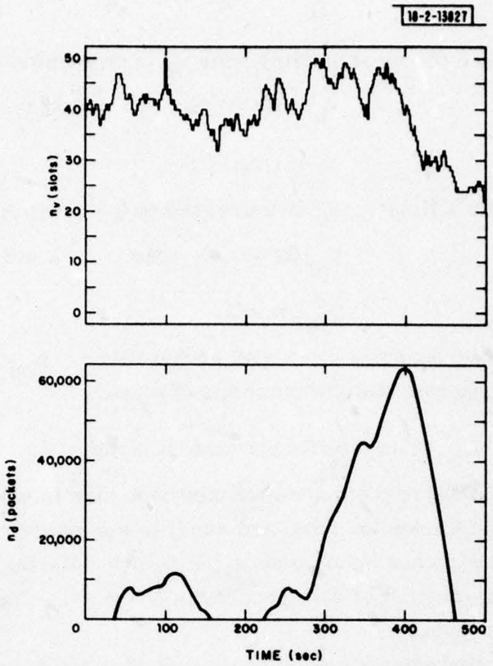


Fig. IV-3. Sample functions of $n_v(t)$ and $n_d(t)$ for $S = 50$, $N = 25$, $\lambda = 0.4\ \text{sec}^{-1}$, $\mu = 0.01\ \text{sec}^{-1}$, and $\Theta = 3400$ packets/sec ($\Theta b = 34$ packets/frame).

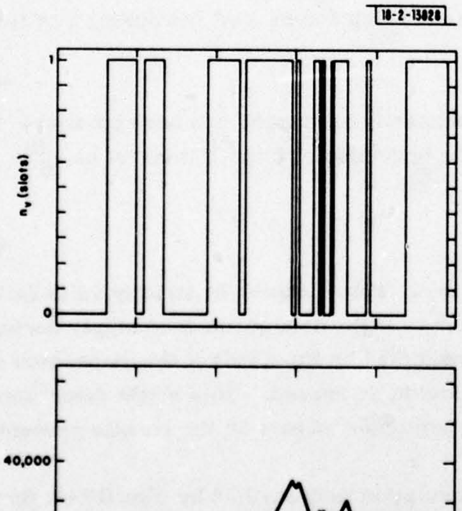


Fig. IV-4. Sample functions of $n_v(t)$ and $n_d(t)$ for $S = 1$, $N = 0$, $\lambda = 0.01\ \text{sec}^{-1}$, $\mu = 0.01\ \text{sec}^{-1}$, and $\Theta = 40$ packets/sec ($\Theta b = 0.4$ packet/frame).

$$W_d = \langle n_d \rangle / \Theta \qquad \text{(IV-8)}$$

where $\langle n_d \rangle$ is a time average of n_d.

3. Simulation Results

a. Analysis of Sample Runs

Sample functions of $n_v(t)$ and $n_d(t)$ are shown in Fig. IV-2 for S = 10, N = 5. The average call holding time is taken as $1/\mu$ = 100 sec, with $\lambda = 0.05 \text{ sec}^{-1}$ so that the offered voice traffic is λ/μ = 5 erlangs. Application of Eq. (IV-1) yields $p_S = 0.018$ and $\overline{n_v} = 4.91$ so that the average number of slots available for data packets is

$$\overline{n_p} = N + S - \overline{n_v} = 10.09 \quad .$$

The data packet arrival rate is $\Theta = 900 \text{ sec}^{-1}$ for an average data utilization factor of

$$\overline{\rho_d} = \Theta b \sqrt{n_p} = 0.89$$

The plots represent 2500 sec of real time, during which approximately 230 voice calls either entered or left the system. The fluctuation of $n_v(t)$ above and below its mean value is apparent. It is also clear that $n_v(t)$ is a highly correlated (in time) random process which exhibits swings of long duration above and below $\overline{n_v}$. This behavior is a consequence of the basic erlangian voice traffic model and is not surprising.

During "idle" periods where n_v is low enough so that $\Theta b < N + S - n_v$, more than enough capacity is available to handle incoming data traffic and an initially empty data queue $n_d(t)$ will remain essentially empty. But, $n_d(t)$ will build up significantly during busy periods where $\Theta b > N + S - n_v$. For this example, a long busy period during which $n_v \geqslant 6$ for about 500 sec begins at about t = 900 sec. During this period the data queue builds up to about 30,000 packets, or 150,000 16-bit words of storage. The data queue is eventually emptied during a subsequent idle period of the voice channel, but then builds up again during the next busy period. The average buffer size during a run of which the first half is depicted in Fig. IV-2 is $\langle n_d \rangle$ = 9200 packets, corresponding to a mean delay of W_d = 10.2 sec.

The behavior depicted in Fig. IV-2 is typical of that exhibited in many similar runs. A segment of another pair of sample functions $n_v(t)$, $n_d(t)$ is shown in Fig. IV-3 for S = 50, N = 25, $\lambda = 0.4 \text{ sec}^{-1}$, $\mu = 0.01 \text{ sec}^{-1}$, Θ = 3400 packets/sec. Here, $\overline{n_v} = 39.2$ and $p_S = 0.02$. There are $S - \overline{n_v} = 10.8$ residual voice channels on the average, and the average data load fills 9 of these residual channels. The plots cover 500 sec of real time and about 390 transitions in n_v. The long busy period ending near t = 400 leads to a maximum data buffer of about 63,000 packets. Average buffer size for this run is $\langle n_d \rangle$ = 12,852 packets, and average delay is $W_d - \langle n_d \rangle/\Theta \approx$ 3.78 sec.

Figure IV-4 illustrates what happens in the extreme case when only one slot per frame (S = 1, N = 0) is available. Here $\lambda = 0.01 \text{ sec}^{-1}$, $\mu = 0.01 \text{ sec}^{-1}$, and $\overline{n_v} = 0.5$ [see Eq. (IV-1)] so that the slot is occupied by voice half the time. The data load Θb is 0.4 packet/slot so that

$$\overline{\rho_d} = \Theta b \sqrt{n_p} = 0.4/0.5 = 0.8 \quad .$$

The data buffer $n_d(t)$ obviously must increase while $n_v = 1$, and these increases must be worked off while $n_v = 0$. For this example, $\langle n_d \rangle$ = 7047 packets and W_d = 176 sec. The effect of the average voice holding time $1/\mu$ should be apparent from this example. If $1/\mu$ were doubled to 200 sec with λ/μ fixed, a typical sample function of $n_v(t)$ would be as shown in Fig. IV-4, except

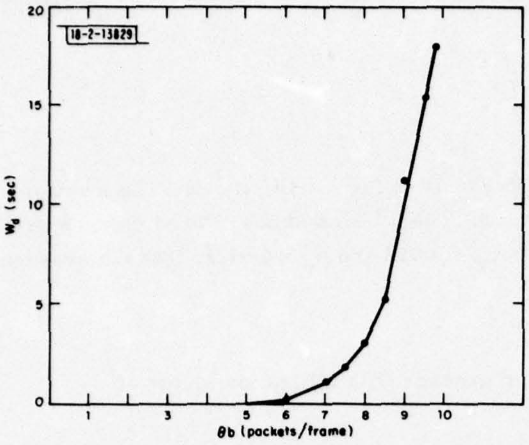


Fig. IV-5. Average data packet delays as function of data load for S = 10, N = 5, λ = 0.05 sec^{-1}, μ = 0.01 sec^{-1}. Each plotted point represents an average of 4 runs, with each run comprising 5000 transitions in $n_v(t)$ and about 50,000 sec of real time.

TABLE IV-1

AVERAGE AND MAXIMUM DATA BUFFER SIZE, IN THOUSANDS OF 16-BIT WORDS, AS FUNCTION OF DATA LOAD

[Each entry represents an average of $<n_d>$ or $(n_d)_{max}$ over 4 runs, with each run comprising 5000 transitions in $n_v(d)$ and about 50,000 sec of real time.]

θb (packets/ frame)	Average Data Buffer (K words)	Maximum Data Buffer (K words)
5	0.002	1.2
6	0.55	42
7	4.16	147
7.5	7.24	189
8	12.3	242
8.25	21.2	380
8.5	22.4	343
8.75	34.5	484
9.0	52.4	798
9.25	54.2	575
9.5	73.9	904
9.75	87.9	846

that twice as much elapsed time would be represented. With Θ fixed, this time scale change would lead to approximately a doubling of the $n_d(t)$ values plotted in Fig. IV-4, and therefore a doubling of $<n_d>$ and W_d. The increased W_d is an obvious consequence of the fact that data packets (arriving when $n_v = 1$) must wait twice as long on the average for the ongoing conversation to terminate.

The previously presented formula[4] for the average data packet waiting time is a function of λ/μ and p_k [see Eq. (IV-1)], but does not vary with μ for fixed λ/μ. The effect of μ on the $S = 1$, $N = 0$ case represents a counterexample to this formula, and changes in μ for fixed λ/μ will have similar effects on the results depicted in Figs. IV-2 and IV-3. Actually, the holding time $1/\mu = 100$ sec is shorter than typically encountered in telephone traffic, so that the results given here on data delays are probably optimistic by about a factor of 2 or 3.

b. Average Data Performance Statistics

Average data packet delays as measured by the simulation are depicted in Fig. IV-5 for $S = 10$, $N = 5$, $\lambda/\mu = 5$, and $\mu = 0.01\ \text{sec}^{-1}$. Each plotted point represents an average of 4 runs, with each run comprising 5000 transitions in $n_v(t)$, or about 50,000 sec of real time. Actually, the average buffer size $<n_d>$ was measured, and Eq. (IV-8) used to obtain W_d. For example, at $\Theta b = 9$, $W_d \approx 10$ sec and $<n_d> \approx 9000$ packets or 45,000 16-bit words. In each run, the maximum as well as the average of n_d was measured. Generally, $(n_d)_{max} \gg <n_d>$, as indicated in Table IV-1. Enough storage must be allocated to handle $(n_d)_{max}$. For example, with $\Theta b = 9.0$, about 800,000 words of storage would be needed to prevent data buffer overflow. Results have been presented (see Fig. 2 in Ref. 4) for data waiting times for the same values of N, S, and λ/μ as utilized in obtaining Fig. IV-5. Assuming b = 10 msec, the earlier results indicate $W_d \approx$ 20 msec for b = 9. This predicted delay is about a factor of 1000 below the simulation results obtained here.

Average data packet delays for $S = 50$, $N = 25$, $\lambda/\mu = 40$, and $\mu = 0.01\ \text{sec}^{-1}$ are shown in Fig. IV-6. Delays are slightly less than in Fig. IV-5 for the same values of $\overline{\rho_d} = \Theta b/(N + S - \overline{n_v})$, partially because the larger value of λ/μ implies that transitions in $n_v(t)$ occur at a faster rate. However, the larger values of Θ imply that buffer sizes are similar to those shown in Table IV-1. Again, for $\overline{\rho_d} > 0.9$, average waiting times on the order of 10 sec are exhibited.

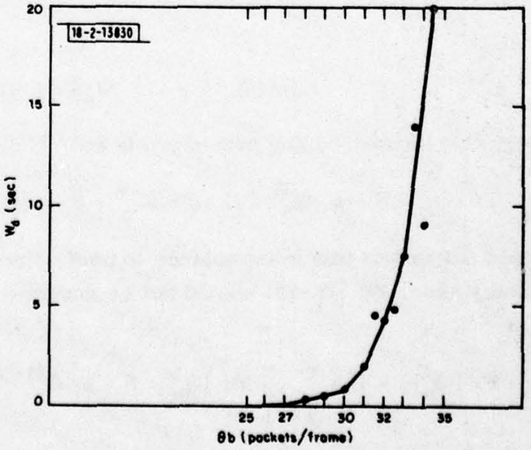


Fig. IV-6. Average data packet delays as function of data load for $S = 50$, $N = 25$, $\lambda = 0.4\ \text{sec}^{-1}$, and $\mu = 0.01\ \text{sec}^{-1}$. Each plotted point represents an average of 2 to 4 runs, with each run comprising 5000 transitions in $n_v(t)$, and about 6500 sec of real time.

4. Discussion

a. Need for Flow Control

The results obtained here led to an important conclusion about statistical multiplexing of voice and data. If one fixes the rate of data traffic high enough to utilize a large portion of the long-term average of the capacity unused by voice, then large data packet queues and delays will occur during periods when voice channel occupancy is high. Unless some sort of control mechanism is provided, data packet queues will become so large as to overflow any reasonable amount of storage in the multiplexer. A flow-control mechanism is needed such that the total flow rate (voice + data) into the multiplexer is kept continuously below the multiplexer's capacity. This implies that data users will sometimes encounter delays in gaining access to the network, but prevents buffer overflow in the network node. Providing such a flow-control mechanism for a single multiplexer is probably fairly straightforward. However, in a large multi-node network, the problem of establishing effective network-wide traffic monitoring and flow control is quite formidable.

b. Discrepancies with Previous Analysis

As noted above, the simulation results presented here differ sharply from those predicted in a previous analysis.[4] Unfortunately, it has not been possible yet to obtain a new set of theoretical results to corroborate the simulation. However, some basic difficulties with the previous analysis can be noted. The analysis at issue concerns data traffic in the "movable boundary case," and is developed in Eqs. (9) through (13) of Ref. 4. Q_n^V and Q_n^D are defined to be the number of occupied voice channels and number of data customers, respectively, in the system just after the nth gate opening (where gate openings are time instants occurring at b = 10-msec intervals when data and voice customers are admitted to the system). In order to derive the distribution of Q_n^D, an expression for (the z-transform of) the conditional probability distribution

$$\Pr [Q_{n+1}^D = i \mid Q_n^D = j, Q_n^V = S - k]$$

is written, and this probability is unconditioned in two stages according to

$$\Pr [Q_{n+1}^D = i] = \sum_k \Pr [Q_n^V = S - k] \sum_j \Pr [Q_n^D = j] \times \Pr [Q_{n+1} = i \mid Q_n^D = j, Q_n^V = S - k] \quad . \tag{IV-9}$$

However, this unconditioning procedure is not valid unless Q_n^D and Q_n^V are independent so that

$$\Pr [Q_n^V = S - k, Q_n^D = j] = \Pr [Q_n^V = S - k] \Pr [Q_n^D = j] \quad . \tag{IV-10}$$

One would not expect this independence to hold, since large Q_n^D tends to be correlated with large Q_n^V. In any case, Eq. (IV-10) should not be assumed <u>a priori</u>. The correct unconditioning relation is

$$\Pr [Q_{n+1}^D = i] = \sum_k \sum_j \Pr [Q_n^V = S - k, Q_n^D = j] \times \Pr [Q_{n+1} = i \mid Q_n^D = j, Q_n^V = S - k] \quad . \tag{IV-11}$$

but it is difficult to proceed beyond Eq. (IV-11) since the required joint probability distribution of Q_n^v and Q_n^D is unknown. In proceeding from Eq. (IV-9), it was assumed that $\Pr[Q_n^v = k] = \Pr[Q^v = k]$ for all n. This assumption, together with the incorrect unconditioning procedure, has the effect of treating Q_n^v as a random process which is independent between adjacent 10-msec frames. Actually, as discussed above and illustrated in Figs. IV-2 through IV-4, the number of active voice calls changes only with call initiation or termination, and is very highly correlated across frame-length time intervals.

Another scheme for analyzing this system is outlined as follows. One can utilize the fact that the pair of state variables Q_n^v, Q_n^D form a stationary (in n) Markov chain with transition matrix

$$P_{v,d|v',d'} = \Pr[Q_{n+1}^v = v,\ Q_{n+1}^D = d \mid Q_n^v = v',\ Q_n^D = d'] \qquad \text{(IV-12)}$$

where the stationary state probabilities

$$\pi_{v,d} = P[Q_n^v = v,\ Q_n^D = d] \qquad \text{(IV-13)}$$

can be obtained (in principle) by solving the linear equations

$$\pi_{v,d} = \sum_{\substack{v'=0,\dots,S \\ d'=0,\dots,\infty}} P_{v,d|v',d'}\, \pi_{v',d'} \qquad \begin{pmatrix} v = 0,\dots,S \\ d = 0,\dots,\infty \end{pmatrix} \quad . \qquad \text{(IV-14)}$$

Then, the data queue distribution

$$\pi_d = \sum_{v=0}^{S} \pi_{v,d}$$

and average data queue size

$$\overline{n_d} = \sum_{d=0}^{\infty} d\pi_d$$

could be determined, and Little's formula could be invoked to obtain average waiting time. Carrying out Eqs. (IV-12) through (IV-14) is straightforward in principle but is complicated algebraically, and an analytic solution to complement the simulation results presented here has not yet been obtained.

C. PACKETIZED VIRTUAL CIRCUIT (PVC) TECHNIQUES

1. Background

PVC techniques combine features of both circuit- and packet-switching technologies to provide a very efficient approach to integrating voice and data in a communications network. While most of the proposals for integrated networks have involved some mixture of circuit-switching techniques for voice and packet switching for data, the PVC approach[5] handles both types of traffic in an essentially uniform fashion, easing the implementation and providing the capability to respond automatically to changes in traffic mix. The PVC network concept attempts to capitalize on the statistical multiplexing advantages inherent in packet technology. At the same time,

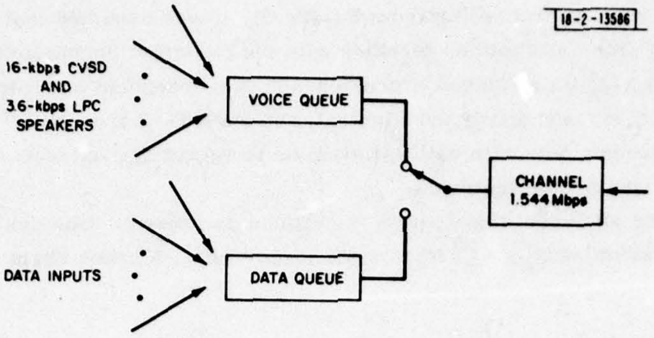


Fig. IV-7. Model of single link in PVC network.

TABLE IV-2

FIXED PARAMETERS IN THE LINK MODEL

Packet size	128 bits
Overhead in packet	32 bits
Data in packet	96 bits
Channel rate	1.544 Mbps
Supervisory traffic and framing	8 kbps
Available channel rate	1.536 Mbps 12,000 packets/sec
Vocoding techniques	CVSD, LPC
CVSD vocoding rate	16 kbps 6 msec between packets
LPC vocoding rate	3.5 kbps 27.5 msec between packets
Voice queue size	70 packets 560 16-bit words 5.83 msec of channel time
Simulation duration	2 min. of channel time

t attempts to overcome some of the efficiency and delay dispersion difficulties associated with ɔure packet networks by utilizing communication link formats and routing conventions associated with digital circuit switching. Since the flow-control mechanisms normally employed in packet networks introduce delays and loss of efficiency which are inappropriate in a network intended to handle a high percentage of voice traffic, the PVC network depends on a flow-control discipline that has been termed "statistical flow control." In addition, the PVC system is designed to take advantage of the on-off statistics of voice traffic to increase its capacity to handle both voice and data.

The PVC approach requires the establishment of a connection from source to destination hosts, fixing most of the packet header information. All packets in the connection follow the same path through the network. The PVC packet header need only contain information identifying it as belonging to a particular connection. Thus, packet overhead is reduced significantly by the use of connections, and short packets can be employed efficiently.

In the PVC scheme, flow control is performed by the assignment of connections to specific links to reduce the probability of internal overloads to values that are small. This permits treating the problems caused by the overloads on an exceptional basis without introducing severe overhead. This new and untried approach to flow control is a vital factor of the PVC network concept.

Packet-speech techniques[9,10] provide a straightforward way to take advantage of silent intervals, which represent more than half of the elapsed time in typical conversational voice transmissions. By using a speech-activity detector and not sending packets when no activity is detected, a packet voice network could expect to handle roughly twice the voice traffic that could be handled if the nominal voice encoding rate had to be handled continuously – that is, by a circuit-switched network with the same channel capacity.

A model of a single link of a PVC network was developed and simulated on the PDP 11/45 computer. The simulation models a population of speakers in conversation, providing a voice load on the system. Data traffic is modeled by a Poisson process. The PVC link model allows the investigation of such variables as buffer space requirements, packet delay, and line utilization as functions of the voice and data loads on the system.

As shown in Fig. IV-7, a single link is modeled as having two distinct input queues for voice and data traffic. When the link is available, a packet is chosen from one queue or the other and transmitted. A summary of the fixed parameters of the model is presented in Table IV-2.

For each run of the simulation, the number of speakers using each vocoder type is specified. Each speaker is determined to be speaking (active) or silent according to distributions of talk spurt and silence distributions obtained from measurements by Brady.[3] When a speaker is determined to be active, he generates packets at a rate characteristic of his vocoding technique. When he is silent, no packets are generated. The model does not attempt to represent the start or end of conversations. When a voice or data packet is generated, it is entered into the respective queue for transmission. The voice queue is finite; when it is filled, new incoming packets are discarded. The maximum size of the data queue is a variable, and is measured for different traffic loads.

A framing strategy is used in which a specified fraction of packet slots in the frame have priority for data. The fraction can vary from 0 to 1, and can be set according to voice and data loads in the node.

Briefly, measurements on the nodal simulation indicate that although, in addition to voice traffic (maximum voice packet utilizations that result in no speech loss – approximately 75 percent), there is sufficient net capacity available for a rate of data traffic that brings the total link utilization (including packet overhead) to 98 to 99 percent, the statistics of voice traffic with absolute priority are such that very large delays in data packet transmissions and unacceptably large queue lengths result. Nonetheless, negligible speech loss and acceptable delays and queue lengths for data can be attained when the data traffic is introduced at rates that result in total link utilizations of 90 to 92 percent and the data traffic is allocated priority for a small fraction (0.1 to 0.2) of a frame.

2. Implementation of a PVC Nodal Processor

The motivation for integrating voice and data in a communications system lies in expected cost savings derived from sharing of transmission and switching facilities, and, to a lesser degree, in the promise of greater flexibility in coping with changing traffic patterns. It is therefore of interest to investigate the implementation of a PVC network nodal switch to provide some means to compare size, complexity, etc. with other known network switches.

a. Tasks of a PVC Nodal Switch

The PVC voice/data network under consideration is assumed to be relatively large and well connected;[11] that is, each of the many nodes is assumed to have approximately 15 to 20 full-duplex 1.544-Mbps link connections and a group of local voice and data users. It is assumed that all the local users can be interfaced to the nodal switch as one of the above full-duplex connections.

The tasks of a PVC nodal switch can be grouped in the following categories:

(1) Store-and-forward processing of voice, data, and network supervisory packets, including management of all the queues.

(2) Setting up and taking down connections; participating in the distributed process that routes new connections from source to destination.

(3) Collecting and distributing network status, for example, remaining unused link capacities; such status is used at the periphery of the network in the "statistical flow control."

(1) Store-and-Forward Processing

A bit count is maintained at each input link for the input bit streams from each modem to locate packet boundaries. When a packet is read in, a check sum is performed and compared with the check-sum bits in the packet. Information in the packet header is used to identify the packet as belonging to a particular connection. Forwarding information is retrieved from local memory and a new header is created. The packet is then output to an appropriate output queue (including one for local users).

The queue management process monitors lengths of all output queues and discards the appropriate packets when certain queues are full. Packets from the queues of each output link must be transmitted to the respective modem. The priorities that were set among the voice, data, and supervisory packets determine which packet type is transmitted during each packet slot.

(2) Routing

When a subscriber requests a virtual connection, a path must be found between source and destination ports which has sufficient uncommitted capacity to handle the requested data rate. If such a path cannot be found, the request will be rejected. In that event, the subscriber has the option of requesting a connection at a lower data rate or waiting until the network is less busy.

Each node attempts to route a connection along a path that has sufficient unused capacity for the connection, and also that minimizes the total number of hops in the path. As connections are established or relinquished, such data are exchanged among all the nodes so that routing is done according to timely information. The routing process must then update the forwarding tables to reflect the new configuration of virtual circuits.

It should be noted that, in a PVC network, only connections are routed and all the packets in a connection follow the same path through the network. In other networks such as the ARPANET, each packet is routed independently. Thus, processing time is not a severe constraint in PVC routing.

(3) Statistical Flow Control

The proposed routing mechanisms in the PVC net, using average rate requirements and statistically obtained safety margins, are intended to control network traffic so that the probability of overload at any point is kept small. However, that probability cannot be made equal to zero; therefore, other mechanisms are required to deal with overloads when they occur. For example:

> Packets can be discarded when queues become excessive.
>
> The peak data rate generated by a subscriber can be limited at the network periphery to the value agreed upon when the virtual circuit was set up.
>
> Average data rates on connections can be monitored and held down to their initial value, or allowed to increase when routing parameters for the connection are adjusted appropriately.

Other than discarding packets, the major flow-control tasks are executed at the periphery of the network. The network nodal switches only provide the necessary information to the host processes at which the flow control is done.

b. A Proposed Configuration for a PVC Nodal Switch

A proposed configuration for a nodal switch in a PVC network is shown in Fig. IV-8. The configuration serves mainly as a strawman for discussion of implementation issues. A significant requirement is that the configuration be modular. A smaller node ought not to have as much hardware as a larger one, and one should be able to easily augment a small nodal switch to handle additional links.

(1) Input and Output Microprocessors

A microprocessor for each input link is postulated to handle incoming packets. A 128-bit packet must be processed every 82.9 μsec. The microprocessor must input the bit stream, locate packet boundaries, and perform a sum check on each packet. The forwarding table at

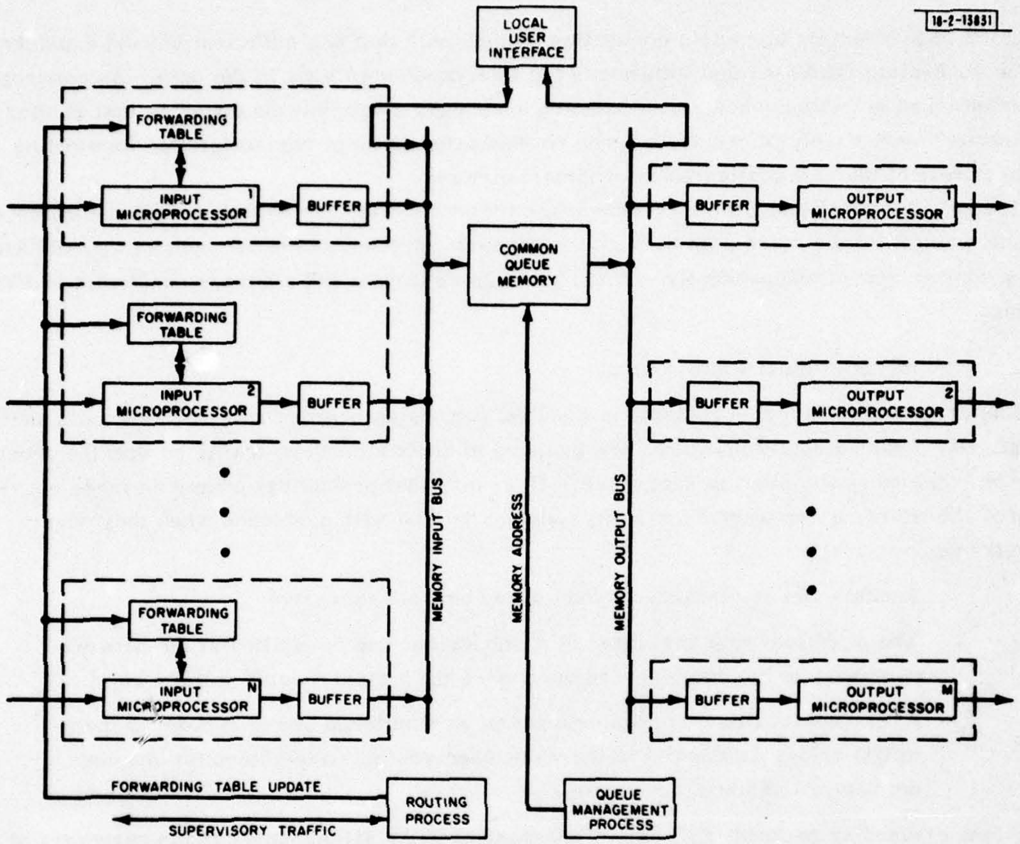


Fig.IV-8. Proposed PVC nodal switch.

each input link has one entry for every connection that is routed through the link. The forwarding address in the header of each packet can be used as an index to the forwarding table. The resulting entry contains information identifying the connection to which the packet belongs, the output link to be used by the packet, a new forwarding address to be used on the output link, and other data concerning the connection type. Forwarding tables might require as many as 48 bits per entry and have approximately 4000 entries per link.[5] When the output link is known, the packet can be transmitted to the appropriate location in the common queue memory. If there is difficulty in completely processing a packet in the allotted time, the problem can be somewhat alleviated by increasing the word length of the microprocessor (for example, to 32 or 64 bits). It appears that a currently available microprocessor (e.g., the AMD 2900 series) with approximately 200,000 bits of external memory can handle the above task.

Similarly, a microprocessor is postulated to output packets. A packet is chosen from one of the link's respective queues according to assigned priorities. Appropriate communications with the queue memory releases the packet slot.

(2) Queue Memory

The queue memory appears to be the most difficult implementation problem. Voice, data, and supervisory packet queues are required for each output link. Each queue must be accessible from all input microprocessors, none of which are operating synchronously. Assuming a voice queue of 70 packets, a data queue of 200 packets, a supervisory queue of 20 packets, and 20 output links, a memory of approximately 750,000 bits is required. Considering a total of forty 1.544-Mbps input and output links, as many as 482,400 packets (3,859,200 16-bit words) are exchanged through this memory every second – a worst-case read/write time of 2.07 μsec per 128-bit packet or approximately 260 nsec per 16-bit word. Clearly, either semiconductor memory or customed interleaved core memory is required.

An advantage to having one memory for all the queues is that the packet buffers can be statistically shared,[12] thus reducing the total amount of memory required. Possible access schemes to the single memory include multiports or input and output busses with appropriate protocols.

A single memory utilizing shared buffer space requires a queue management process that is not overly complex but operates at high speeds. Practical buffer-sharing schemes require a minimum allocation of buffers for each queue and also maximum lengths for all queues.[13] Assuming that the queues will not occupy fixed locations in the memory, the queue management process must keep track of all packet locations, empty buffer locations, and the number of packets in each voice, data, and supervisory packet queue. In addition, the management process must be able to provide an address of an empty buffer for an incoming packet, add it to the appropriate queue, and also provide the address of an outgoing packet, remove it from the appropriate queue, and add the packet's buffer to the list of empty buffers. If 40 input and output lines are operating at full capacity, an address must be provided to the common queue memory every 2.07 μsec. The time constraint of such a task appears too great for a single minicomputer such as the PDP 11 or the Nova. A multiprocessor configuration or a fast processor such as the Lincoln Digital Signal Processor (LDSP) would be able to handle the queue management link.

An alternative to the common queue memory is to have individual first-in, first-out (FIFO) buffers for each voice, data, and supervisory queue. Access to the queues from the input microprocessors is handled by a switch that is comprised of two back-to-back multiplexers. One input multiplexer (approximately 20 to 1) accesses each of the input microprocessors once per packet time (every 82.9 μsec for a 1.544-Mbps link). Each packet has appended to it 6 bits that select to which queue (FIFO buffer) the output multiplexer (approximately 1 to 60) is switched. A relatively simple queue management process must still monitor the number of packets in each queue to prevent overflows. Such a queue memory configuration, however, does not allow for statistical memory sharing.

REFERENCES

1. Network Speech Processing Program Semiannual Technical Summary, Lincoln Laboratory, M.I.T. (31 March 1977), DDC AD-A045454.

2. G. Coviello and P. A. Vena, "Integration of Circuit/Packet Switching in a SENET (Slotted Envelope NETwork) Concept," Natl. Telecommunications Conf., New Orleans, December 1975, pp. 42-12 to 42-17.

3. P. T. Brady, "A Technique for Investigating On-Off Patterns of Speech," Bell Syst. Tech. J. 44, 1-27 (1965).

4. M. J. Fischer and T. C. Harris, "A Model for Evaluating the Performance of an Integrated Circuit- and Packet-Switched Multiplex Structure," IEEE Trans. Commun. COM-24, 195 (1976).

5. J. W. Forgie and A. G. Nemeth, "An Efficient Packetized Voice/Data Network Using Statistical Flow Control," Proc. Natl. Computer Conf., Chicago, 12-15 June 1977.

6. A. G. Nemeth, "Behavior of a Link in a PVC Network," Technical Note 1976-45, Lincoln Laboratory, M.I.T. (7 December 1976), DDC AD-A036370/5.

7. P. Demko, "Measurements of Voice and Data Queue Behavior in a PVC Network Link," Technical Note 1977-37, Lincoln Laboratory, M.I.T. (25 August 1977), DDC AD-A047101.

8. L. Kleinrock, Queueing Systems, Volume 1: Theory (Wiley, New York, 1975).

9. J. W. Forgie, "Speech Transmission in Packet-Switched Store-and-Forward Networks," Proc. Natl. Computer Conf., Anaheim, California, 19-23 May 1975, pp. 137-142.

10. __________, "Subjective Effects of Anomalies in Packetized Speech," J. Acoust. Soc. Am. 60, Suppl. 1, 5109 (1976) (abstract).

11. "SENET-DAX Study – Final Report" Volume 2, GTE Sylvania, Inc., Electronic Systems Group, Eastern Division (25 June 1976).

12. F. Kamoun, "Design Considerations for Large Computer Communications Networks," UCLA Ph. D. Dissertation, March 1976.

13. P. Kermani and L. Kleinrock, "Analysis of Buffer Allocation Schemes in a Multiplexing Node," Computer Science Dept. UCLA (November 1976).

APPENDIX A
TELECONFERENCING QUESTIONNAIRES

Introduction to
Teleconferencing Questionnaire #1

A Review of Experimental Conditions to Date

Up to this point, you have participated in at least 10 experimental teleconferencing sessions. During the early sessions, you solved "car pool" problems in four-person groups, somewhat later, in eight person groups, and, very recently, in twelve-person groups. In addition to gaining experience with conferences of different sizes, you have gained experience with two basically different types of telephone systems. One of these, the "analog bridge," is very similar to the common telephone system. The system permits any number of simultaneous speakers to be heard by each other and by all listeners. The second, or "voice control" system, is considerably different from the analog bridge in a number of respects. From the listener's point of view, one of the most prominent of these is that only one speaker can be heard, though several might be attempting to talk. When one speaker has finished, a second may then be heard, though the listener may be aware that early portions of the second speaker's message have been lost.

The Purpose of This Questionnaire

Your perceptions of the ease or difficulty with which conferences can be conducted and problems solved within groups of different sizes using different systems are critical to successful evaluation of various teleconferencing arrangements. Your preferences, if any, among the alternatives are also important.

From time to time, we will ask you to fill out a short questionnaire regarding your perceptions, preferences, and comments. The data provided by you will be used in conjunction with other measures of conference performance with which you are familiar (e.g., solution time, solution quality, tape recordings of the discussions, computer data on the functioning of the phone systems, etc.) in our report of the experimental trials.

Please read and answer all of the questions carefully. When you are finished, put your name in the appropriate space and do either of the following:

- Return the form to Chris.

or

- Keep the form handy and bring it to the next experimental session.

Thank you very much for your continuing cooperation.

BBN/Lincoln

TELECONFERENCING
QUESTIONNAIRE #1

PLEASE READ ITEMS CAREFULLY AND COMPLETELY BEFORE ANSWERING

1. Immediately below is a set of five conferencing conditions, each member of which is described by the telephone system employed, the number of conferees and the number of commuters involved in the car pool problem to be solved. You have already served as a subject in each of these conditions.

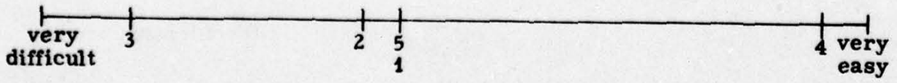
Index No.	No. of Conferees	Telephone System	No. of Commuters
1	12	Analog Bridge	12
2	12	Voice Control	12
3	8	Analog Bridge	8
4	8	Voice Control	8
5	4	Analog Bridge	6

We ask you to imagine that you will shortly be required to be a subject during a repetition of this set of experimental conditions. This time, however, you are considerably more experienced and have a better perspective on the conferencing situations. As a result, you are able to make an estimate of the rank order of difficulty of the conditions and, further, to make a judgment about how much more difficult or easy one condition will be than another. In addition, you recognize that, as a result of your accumulating experience, your current perception of the relative difficulty of different conditions may not be the same as it was when you were less sophisticated.

It is this current perception, this feeling that you now have about how the conditions would be distributed with respect to difficulty if you were to encounter them again, that we want you to indicate below. Note that you are not being asked to attempt to remember how difficult the conditions seemed at the time, but rather how they now seem in advance of a repetition.

Below, there is a line on which we want you to make your judgments of the relative difficulties of the conference conditions. One end of the line is labeled "very difficult," the other, "very easy." Indicate your judgment of the difficulty of each condition by marking the line at the appropriate point and identifying the mark with the index number associated with that condition in the above Table. Indicate conditions of equal difficulty by listing associated index numbers in a column below the mark

EXAMPLE:



In this example, a fictitious subject has indicated the belief that condition 3 is quite difficult, that 2 is considerably less difficult than 3 but slightly more difficult than 5 and 1, which are equal. In this subject's view, condition 4 is very much easier than any of the conditions.

Now it is your turn.

very difficult — very easy

2. Describe, as best you can, the reason(s) why you distributed the conditions as you did.

3. What percentage of the remaining subjects do you believe will distribute the conditions in the same rank order you have? (NOTE: This question concerns order alone, not the distances between marks). Check one.

○ 0–20%　○ 41–60%　○ 81–100%

○ 21–40%　○ 61–80%

4a. How frequently do you believe you can identify conference participants on the basis of the sounds of their voices when using the analog bridge system?

○ almost always

○ frequently

○ infrequently

○ almost never

4b. How frequently do you believe you can identify conference participants on the basis of the sounds of their voices when using the voice control system?

○ almost always

○ frequently

○ infrequently

○ almost never

4c. How important is it to you that you know who is speaking at a given time?

○ very important

○ important

○ not very important

○ very unimportant

5a. Assume that, at some future time, an effort was to be made to determine if a conference involving the solution of car pool could be conducted more efficiently by employing a chairman. The primary task of this chairman would be to eliminate interruptions of one speaker by others. Assume that you were the person chosen. On which of the two systems would you prefer to carry out that role?

○ Analog Bridge

○ Voice Control

○ No Preference

5b. Please explain the reason(s) for the alternative selected above.

PLEASE SIGN YOUR NAME______________________________

TELECONFERENCING
QUESTIONNAIRE #2

PLEASE READ ITEMS CAREFULLY AND COMPLETELY BEFORE ANSWERING

1. Below is the set of five conferencing conditions you rated for relative difficulty three weeks ago, and a copy of your rating form:

CONFERENCE CONDITIONS

Index No.	No. of Conferees	Telephone System	No. of Commuters
1	12	Analog Bridge	12
2	12	Voice Control	12
3	8	Analog Bridge	8
4	8	Voice Control	8
5	4	Analog Bridge	6

YOUR EARLIER RATING

Today you have had experience with a second set of analog bridge and voice control conditions. In this latter set, a delay typical of that which would be experienced during communications involving a satellite was introduced. We would now like you to merge your impressions of these systems with those portrayed in your earlier rating.

Since you have only been in eight-person conferences using the delay conditions, we will eliminate the 12-person and 4-person conditions, leaving the following set for you to judge.

Index No.	No. of Conferees	Telephone System	No. of Commuters
3	8	Analog Bridge	8
4	8	Voice Control	8
6	8	Analog Bridge with Delay	8
7	8	Voice Control with Delay	8

As before, base your judgment on your impression of how the conditions would rank if we were to repeat the experiments in the future.

NOTE: There is no need to maintain either your earlier rank order of Index Nos. 3 and 4 or your original spacing. If, in your judgment, either order or spacing has changed in the light of Index Nos. 6 and 7, make the change(s) in the rating.

YOUR NEW RATING

____

very difficult ... very easy

2. Distribute Index Nos. 3, 4, 6, and 7 on the line below in accord with the relative frequency with which you, as a speaker, feel you would be heard and understood by the rest of the conferees in a future repetition of the experiments.

____

always ... never

3. Distribute Index Nos. 3, 4, 6, and 7 on the line below in accord with your judgment of the relative ease of interrupting a given speaker when you, as a listener, have something to say.

____

very difficult to interrupt ... very easy to interrupt

4. Distribute Index Nos. 3, 4, 6, and 7 on the line below in accord with your judgments of the relative frequencies with which you can identify speakers on the basis of the sounds of their voices.

____

always ... never

INSTRUCTIONS FOR SENTENCES

Place a mark under each sentence in the place which expresses your opinion.

You can mark off the line at either end to express an extreme opinion (but make sure we can find it).

The center position is intended to represent neutrality, although several descriptive words are used. If you mark on the centerline, it means you feel equally about both ends of the scale.

Treat each sentence independently. Do not try to make answers match.

Work quickly – read the whole sentence and mark the line.

Do the sentences in order. Do not skip any.

Name____________________ Date________________ Time ____________________

This place is	fine	adequate	terrible	to work in.
Speech was	easy	normal	difficult	to understand.
We each get	little	enough	much opportunity	to participate.
This problem was	easier	same	harder	than others.
The system produced	few	average	many	spurious noises.
There were	no	usual	many	repeat requests.
The handset, buttons, etc. are	easy	normal	hard	to manage.
People talked	rarely	usual	often	at once.
Quality degradation was	less	usual	more	than on other systems.

I had to speak	softer	same	louder	than usual.
This system requires	little	usual	much	fuss to use.
This system changed	no	some	all	voices.
Communication on this system was	easier	same	harder	than on other systems.
I experienced	no	average	great	difficulty being understood.
For this problem, I	like	don't mind	dislike	this system.
This system is	better	same	worse	than most others.
I missed	few	some	many	words.
This system is	excellent	average	terrible	in overall quality.

Name____________________________

Date______________ Time________

Put a mark on the line if the system made any or all voices sound:

_____ as good as on my office telephone

_____ clicky

_____ cutoff

_____ distorted

_____ fuzzy

_____ garbled

_____ monotonic

_____ muffled

_____ nasal

_____ normal

_____ produced by machine

_____ squeaky

_____ unintelligible

_____ unreal

_____ - - - - - - - - - - - - - -

_____ - - - - - - - - - - - - - -

Use the lines and space at the bottom to indicate any qualities of speech you heard not on the list.

APPENDIX B
AUDIT TRAILS AND STATISTICAL PACKAGE

EXPLANATION OF CODES USED IN THIS APPENDIX

This appendix contains a sample output generated by the data reduction program upon the completion of a conference experiment. The first part of the output is an audit trail showing the progress of the conference as a function of time. This particular conference involved twelve participants and lasted for 770 sec. It used the SI voice-controlled signal-selection technique. Time is represented in horizontal bands with tick marks every 10 sec. Within each band are rows for each participant, which are labeled by the columns of numbers on the left and right margins. The row labeled "20" is used to note events marked by the experimenters, such as the actual starting time of the conferencing problem, by recording a short "beep" on the audio tape. Each character in a row indicates the "state" of the participant during a 1-sec interval. If he was the selected "speaker" during the interval, a "|" appears. If he was the "interrupter," a "-" appears. If the participant changed between speaker and interrupter or vice versa, "+" appears. If he produced signal energy above the speech activity threshold but was not selected as either speaker or interrupter, a "o" appears. If he was silent, no mark appears. The numbers below the tick marks at the bottom of each band show the time since the startup of the conferencing program.

Following the audit trail, a series of summary statistics are printed. Each is identified by a title. The statistics are based on samples taken every 20 msec from each participating phone line. Each sample is categorized as belonging to one of eight states which have the following meanings:

State	Meaning
0	No speech detected
1	Speech above threshold detected but channel not assigned
2	"Speaker" assigned but currently not speaking
3	"Speaker" assigned and currently speaking
4	"Interrupter" assigned but currently not speaking
5	"Interrupter" assigned and currently speaking
6	"Speaker" and "interrupter" assigned but neither speaking *(not a meaningful category at this time)*
7	"Speaker" and "interrupter" assigned and both speaking

```
Filename: ts5
number of header words = 4
start data coll at Fri Oct 14 09:18:26 1977
trailing threshold = 250

from beginning to end

---------------------------------------------------------------------------------

audit chars:
        |       speaker
        -       interrupter
        +       both speaker and interrupter
        o       above threshold, not speaker or interruyter
        (space) none of the above
Each column is 50 * 20ms

 1:                                                         |||            :1
 2:          ||----               -- --          || o          +|          :2
 3:           o                         o                          ||  +|  :3
 4:                                                ||||             ||||   :4
 5:           oo ||                                     |||                :5
 6:           --o ||||||||   o|                        +++-+|-+|+||     || :6
 7:           ||||--   o -+|||  ||     ---    +||                          :7
 8:                   ---o    || ||||         --      --                   :8
 9:                        --- ||      o||      --  - ||  ||               :9
10:           oo         ||                 ||| ||      -                  :10
11:          | +| o ---   --        +|||||||                               :11
12:                                                                        :12
20:                                                                        :20
             !         !         !         !         !         !         !
         10.00     20.00     30.00     40.00     50.00     60.00     70.00

  136.86  beep on
  137.12  beep off at addr 0,7126 (oct) in filo
 1:                                                                        :1
 2:  ||||:|+||                                                             :2
 3:                             |||                                        :3
 4:                                +|                                      :4
 5:                                     ||      ---           ||  |||||||| :5
 6: |                    ||||||| |||| || |||||||||||||||||-+|||  || --     :6
 7:                                        ||                              :7
 8:           |||                             ||          ||               :8
 9: ||    +|           |||                      ||            -+| --       :9
10:             --                                +|       |               :10
11:             ||||||                              +|                     :11
12:                                                    ||                  :12
20:                                                                   oo   :20
             !         !         !         !         !         !         !
         80.00     90.00    100.00    110.00    120.00    130.00    140.00

 1:           ||||||||-+||||||                                             :1
 2:                                                                        :2
 3:                                                                        :3
 4:                                                                        :4
 5: |||||||                                                                :5
 6:                                                                 |      :6
 7:                                                                        :7
 8:                                                                   |||| :8
 9:                                          ||||||| |||:|                 :9
10:    --                                                   ||||||||+--    :10
11:                   ||     |||||| ||||||                                 :11
12:                                                                        :12
20:                                                                        :20
             !         !         !         !         !         !         !
        150.00    160.00    170.00    180.00    190.00    200.00    210.00

 1:                                                                        :1
 2:                                                                        :2
 3:                                                  |||||||||             :3
 4:                                                             +||||||||| :4
 5:                                                                        :5
 6:                         ||||||| ||||||                                 :6
 7:             I   IIIIIII                                                :7
 8: II IIIIII                                                              :8
 9:                                                             I          :9
10:                                                                        :10
11:                                                                        :11
12:                                        IIIIIIII                        :12
20:                                                                        :20
             !         !         !         !         !         !         !
        220.00    230.00    240.00    250.00    260.00    270.00    280.00
 1:                                        --      IIIIII                  :1
 2:         IIIIIIIII IIIII         III                                    :2
 3:                                                                        :3
 4:  IIII                          --+I    IIIIIII                -        :4
 5:        '                                                               :5
 6:                          IIIIIIII                                      :6
 7:         =                                                              :7
 8:                                     II                  III  II        :8
 9:                                       II                               :9
10:                        .                                     I         :10
11:                                                              IIII      :11
12:                                                                   IIII :12
20:                                                                        :20
             !         !         !         !         !         !         !
        290.00    300.00    310.00    320.00    330.00    340.00    350.00
 1:                                                                        :1
 2:                                           I-- IIII    +II              :2
 3:                                                             -          :3
 4:                             III                    II-+                :4
 5:                         II             II  IIII-- IIII                 :5
 6:                                 IIII                    +II    II IIII :6
 7:                                                        o               :7
 8:                           II                                           :8
 9:                IIIIIIIII   +                             --      --    :9
10:                                                                        :10
11:                                                       +I    II  III -- :11
12: IIIIIIIIIIIII                                                          :12
20:                                                                        :20
             !         !         !         !         !         !         !
        360.00    370.00    380.00    390.00    400.00    410.00    420.00
 1:                             -                     IIIIIIII  ----       :1
 2:                                                                        :2
 3: - -+II         III      II    IIII                                     :3
 4:              II                    II--                                :4
 5:             +I        o              I                    II-IIIIII  I :5
 6: III II       o       oo--    --       IIII    IIII                      :6
 7:                               o                                        :7
 8:        oo        +I I+-   IIII  -+                                     :8
 9:                   II          -       o                                :9
10:    o   --                    II                            +II     II  :10
11:   II    IIIII-       IIII                                              :11
12:                                                                        :12
20:                                                                        :20
             !         !         !         !         !         !         !
        430.00    440.00    450.00    460.00    470.00    480.00    490.00
 1:                         --                                             :1
 2:                         IIII    IIII  IIIIII          -- --            :2
 3:                                                     !                  :3
 4:                                                                        :4
 5: II       o     II   II           --               --IIIIoo         - I :5
 6:                                    III     +I IIIII  II -+IIIIIIIIII    :6
 7:                                                                        :7
 8:  +IIIIIII+IIIII-+III-II                                    o           :8
 9:                       III      II                                      :9
10:        -+             --   IIIII            +I  --+I                   :10
11:                                                         +III           :11
12:                                                                        :12
20:                                                                        :20
             !         !         !         !         !         !         !
        500.00    510.00    520.00    530.00    540.00    550.00    560.00
 1:                                          ||||
 2:                                                                          :1
 3:            ||                                                            :2
 4:                           ||           ||||  +||                         :3
 5: ||||| ||||          |||||o|||                                      |||  :4
 6:      --                                                        ---       :5
 7:                                                           -+|||||||      :6
 8:       |      ||||||||   ||     |||||                                     :7
 9:                                                                          :8
10:           +||            --              --                              :9
11:                 --                                                       :10
12:                                                |||||||||||||||           :11
20:                                                                          :12
           !         !         !         !         !         !         !     :20
      570.00    580.00    590.00    600.00    610.00    620.00    630.00
 653.02  wrap at addr 0,33354 (oct) in file
 1:
 2:                                                                          :1
 3:                                           ||||||||                       :2
 4: |  +|                                                                    :3
 5:            ||         ||                                              +  :4
 6: -+||       ||         ||          ||  ||  ||        ||||||||||||||  |||| :5
 7:                                                                          :6
 8:  -                             ||  -+|                                   :7
 9:                                                               ||||  --   :8
10: ||-+ |||||    --+||                                                      :9
11:               |||                                                        :10
12:                                                                          :11
20:                                                                          :12
           !         !         !         !         !         !         !     :20
      640.00    650.00    660.00    670.00    680.00    690.00    700.00
 722.00  end at addr 0,35510 (oct) in file
 1:
 2:                                                                          :1
 3:                                                                          :2
 4: ||                                                                       :3
 5:     -- +|||||||||||||||                                                  :4
 6:                                                                          :5
 7:                                                                          :6
 8:         --                                                               :7
 9:    |||||                                                                 :8
10:  |||                                                                     :9
11:                                                                          :10
12:                                                                          :11
20:                                                                          :12
           !         !         !         !         !         !         !     :20
      710.00    720.00    730.00    740.00    750.00    760.00    770.00

--------------------------------------------------------------------------------

last time stamp at    722.00
  3730 time stamp words in file
maxdur -    584.89 seconds at addr 35510 (octal)

--------------------------------------------------------------------------------

        Counts of changes by phones into these states
        0      1      2      3      4      5      6      7 total
pho
  1    30      2     62     62      8      8      0      1   173
  2    37      8    132    131     22     23      0      1   354
  3    20      4     42     42      6      8      0      0   122
  4    31      6    102    102     13     18      0      3   275
  5    71     13    211    205     15     16      0      1   536
  6    68     14    330    330     27     35      0      2   806
  7    20      7     27     27     14     16      0      0   111
  8    70     14    126    125     19     22      0      3   379
  9    38      8     82     81     15     17      0      2   243
 10    40      9     83     87     25     29      0      5   293
 11     40      8   171   171    21    24     0     1   436
 12     10      1    62    62     0     0     0     0   135
 20      1      1     0     0     0     0     0     0     2

tot    476     95  1435  1429   185   216     0    19  3835

----------------------------------------------------------------------------

       Counts of durations in various states

dur (ms)         0      1      2      3      4      5      6      7

   0- 180       80     86    816   1134     94    107      0     19
 200- 380       44      9    219    272     21     27      0      0
 400- 580       32      0    400     20     53      2      0      0
 600- 780       24      0      0      2      9      0      0      0
 800- 980       18      0      0      0      3      0      0      0
1000-1180       10      0      0      0      2      0      0      0
1200-1380        9      0      0      0      1      0      0      0
1400-1580        6      0      0      0      0      0      0      0
1600-1780       13      0      0      1      1      0      0      0
1800-1980       15      0      0      0      1      0      0      0
2000-2180        4      0      0      0      0      0      0      0
2200-2380        5      0      0      0      0      0      0      0
2400-2580        1      0      0      0      0      0      0      0
2600-2780        3      0      0      0      0      0      0      0
2800-2980        1      0      0      0      0      0      0      0
3000-3180        8      0      0      0      0      0      0      0
3200-3380        5      0      0      0      0      0      0      0
3400-3580        7      0      0      0      0      0      0      0
3600-3780        1      0      0      0      0      0      0      0
3800-3980        1      0      0      0      0      0      0      0
4000-4180        2      0      0      0      0      0      0      0
4200-4380        4      0      0      0      0      0      0      0
4400-4580        4      0      0      0      0      0      0      0
4600-4780        3      0      0      0      0      0      0      0
4800-4980        1      0      0      0      0      0      0      0
5000-5180        4      0      0      0      0      0      0      0
5200-5380        2      0      0      0      0      0      0      0
5400-5580        2      0      0      0      0      0      0      0
5600-5780        4      0      0      0      0      0      0      0
5800-5980        2      0      0      0      0      0      0      0
6000-6180        1      0      0      0      0      0      0      0
6200-6380        1      0      0      0      0      0      0      0
6400-6580        0      0      0      0      0      0      0      0
6600-6780        4      0      0      0      0      0      0      0
6800-6980        3      0      0      0      0      0      0      0
7000-7180        2      0      0      0      0      0      0      0
7200-7380        2      0      0      0      0      0      0      0
7400-7580        2      0      0      0      0      0      0      0
7600-7780        1      0      0      0      0      0      0      0
7800-7980        1      0      0      0      0      0      0      0
8000-8180        1      0      0      0      0      0      0      0
8200-8380        0      0      0      0      0      0      0      0
8400-8580        3      0      0      0      0      0      0      0
8600-8780        1      0      0      0      0      0      0      0
8800-8980        1      0      0      0      0      0      0      0
9000+...       151      0      0      0      0      0      0      0

----------------------------------------------------------------------------

       Number of seconds each phone is in each state
pho        0         1         2         3         4         5         6         7

  1   693.22      0.06     18.84      5.58      3.94      0.34      0.00      0.02
  2   672.02      0.32     27.14     14.40      5.26      2.34      0.00      0.02
  3   701.00      0.18     13.20      4.36      1.60      1.15      0.00      0.00
  4   677.74      0.34     22.80     16.22      3.06      1.78      0.00      0.05
  5   637.46      1.32     46.26     29.32      6.34      1.29      0.00      0.02
  6   606.06      1.04     57.80     47.32      5.84      3.90      0.00      0.04
  7   705.30      0.66      8.58      3.22      2.68      1.55      0.00      0.00
  8   661.58      1.06     37.22     12.52      7.06      2.50      0.00      0.06
  9   680.50      0.62     26.06      7.38      5.78      1.62      0.00      0.04
 10   678.34      0.34     21.24     10.42      8.04      3.02      0.00       0.1
 11  659.90    0.80   31.86   22.80    4.42    2.20    0.00    0.02
 12  700.26    0.02   15.22    6.50    0.00    0.00    0.00    0.00
 20  721.74    0.26    0.00    0.00    0.00    0.00    0.00    0.00
--------------------------------------------------------------------------------

Total times: n phones simultaneously > threshold (bit0-1)

#ph     time (seconds)

  0      521.96
  1      190.82
  2        8.00
  3        1.06
  4        0.16
  5+       0.00

sum      210.64

--------------------------------------------------------------------------------

Total time spent by each speaker speaking (bit1-1)
spkr  time (secs)  (each x = 2 secs, rounded up)
  1        24.44  xxxxxxxxxxxxx
  2        41.56  xxxxxxxxxxxxxxxxxxxxx
  3        18.06  xxxxxxxxxx
  4        39.08  xxxxxxxxxxxxxxxxxxxx
  5        75.60  xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
  6       105.16  xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
  7        11.80  xxxxxx
  8        49.80  xxxxxxxxxxxxxxxxxxxxxxxxx
  9        33.48  xxxxxxxxxxxxxxxxx
 10        31.76  xxxxxxxxxxxxxxxx
 11        54.58  xxxxxxxxxxxxxxxxxxxxxxxxxxxx
 12        21.72  xxxxxxxxxxx
 20         0.00

--------------------------------------------------------------------------------

        # of times speaker spoke (bit1-1) this many ms
spkr   0- 980  1000-1980  2000-2980  3000-3980  4000-4980  5000-5980  6000+....
  1       12         10          1          9
  2       10         11          1          2          0          0          0
  3        5         10          1          0          1          1          0
  4        8         13          3          1          0          0          0
  5       17         23          8          1          0          1          0
  6       11         25          9          3          2          0          0
  7        9          4          0          0          2          3          1
  8       35         13          1          2          0          0          0
  9       16          6          4          1          0          0          0
 10                                                    0          0          0
```

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

19 REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER 18 ESD-TR-77-337✓	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* 6 Network Speech Processing Program		5. TYPE OF REPORT & PERIOD COVERED Annual Rep[redacted]t 9 1 Octo[redacted]976 – 30 Sep[redacted]77
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) 10 Melvin A. Herlin [redacted] Theodore Bially		8. CONTRACT OR GRANT NUMBER(s) 15 F19628-76-C-0002✓
9. PERFORMING ORGANIZATION NAME AND ADDRESS Lincoln Laboratory, M.I.T.✓ P.O. Box 73 Lexington, MA 02173		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS ~~Program Element No. 33126K~~
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Communications Agency 8th Street & So. Courthouse Road Arlington, VA 22204		12. REPORT DATE 11 30 Sep[redacted]77
		13. NUMBER OF PAGES 84 83p.
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)* Electronic Systems Division✓ Hanscom AFB Bedford, MA 01731		15. SECURITY CLASS. *(of this report)* Unclassified 12
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

network speech processing
secure voice conferencing
speech algorithms
integrated networks
packetized speech
human factors
speech encoding
teleconferencing

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

This report covers work performed during FY 1977 on the DCA Network Speech Processing Contract. Three general areas of work are reported in this document: secure voice conferencing, speech algorithm studies, and bandwidth efficient communications.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

207 650